FUNCTION DALAM PHP

Seperti bahasa pemrograman yang lain, PHP mendukung modularity program. Umumnya modularity suatu program diwujudkan dalam bentuk suatu function.

Suatu fungsi dapat melakukan suatu pengeksekusian sekumpulan kode yang mempunyai kegunaan/fungsi tertentu. Suatu function dapat bekerja berdasarkan parameter-parameter yang diinputkan kepadanya. Suatu function juga dapat menghasilkan suatu nilai berdasarkan proses yang ada di dalamnya.

Ada beberapa hal-hal yang penting dan harus diketahui mengenai function yaitu :

a. Pendeklarasian Function dan Pemanggilan Function
b. Nilai Kembalian (Return Value)
c. Variable Function
d. Membuat Library Function
e. Built-in Function (Function internal dari PHP)

Pendeklarasian Function dan Pemanggilan Function

Pendeklarasian function dapat dilakukan di lokasi manapun dalam file php. Sebuah pendefinisian function umumnya terdiri dari 3 bagian yaitu :

1. Nama function
2. Daftar parameter yang dipisahkan dengan tanda koma, dan diapit oleh tanda kurung.
3. Isi function yang diapit oleh tanda kurung kurawal ({ })

Bentuk umum dari sebuah function PHP adalah sebagai berikut :

```
function function_name ($parameter1, $parameter2, ..., $parameterN) {
        Isi function;//Kode program ditulis disini.
        return  nilai; // Sifatnya opsional.
}
```

Aturan-aturan dalam pembuatan function adalah :

1. Aturan penamaan function mirip dengan penamaan variable. Terdiri dari huruf, angka dan underscore (_). Nama function hanya boleh dimulai dengan huruf atau dengan underscore.
2. Parameter sifatnya tambahan. Boleh saja function tanpa memiliki parameter.
3. Function boleh memiliki nilai kembalian (return value) ataupun tidak.
4. Variabel yang dideklarasikan dalam function hanya berlaku dalam function saja.

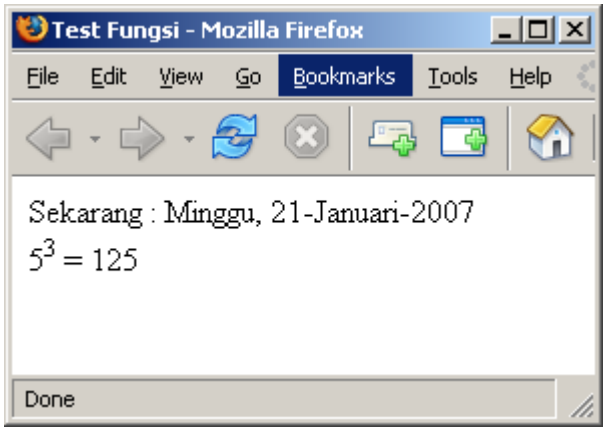
Untuk dapat memanggil function tersebut agar dieksekusi adalah dengan memanggil nama functionnya disertai dengan parameternya (jika mewajibkan ada parameter).

Contoh pendeklarasian fungsi adalah sebagai berikut :
Nama File : func1.php

```
1  <html>
2  <head><title>Test Fungsi</title></head>
3  <body>
4  <?php
5          function info_tanggal()
6          {
```

```
 7              $hari=array("Minggu","Senin","Selasa","Rabu","Kamis","Jum'at","Sabtu");
 8              $bulan=array("","Januari","Februari","Maret","April","Mei","Juni","Juli",
                             "Agustus","September","Oktober","November","Desember");
 9              $tgl=getdate();
10              echo $hari[$tgl['wday']].", $tgl[mday]-".$bulan[$tgl['mon']]."-$tgl[year]";
11          }
12          function pangkat($x, $n)
13          {
14              return pow($x,$n);
15          }
16  ?>
17  Sekarang : <?php info_tanggal();?><br>
18  5<sup>3</sup> = <?php echo pangkat(5,3);?>
19  </body>
20  </html>
```

Jika dieksekusi, maka akan menghasilkan tampilan seperti di bawah ini.



Keterangan :

Baris	Keterangan
5 - 11	Pendeklarasian fungsi **info_tanggal()** yang akan menampilkan informasi tanggal sekarang. Fungsi ini hanya akan menampilkan informasi tanggal dan tidak mereturnkan suatu nilai apapun.
7	Pendeklarasian array **$hari** yang merupakan array yang berisi nama-nama hari, dimulai dari index ke-0 untuk hari Minggu.
8	Pendeklarasian array **$bulan** yang merupakan array yang berisi nama-nama bulan, dimulai dari index ke-1 untuk bulan Januari.
9	Pemanggilan fungsi **getdate()** yang berguna untuk mengambil tanggal sekarang dari komputer dan disimpan di variable **$tgl**. Fungsi getdate menghasilan/mereturnkan sebuah array yang berisi informasi waktu sekarang. Untuk lebih jelas mengenai fungsi ini, silahkan cari di PHP Manual pada kategori Date Time Function.
10	Menampilkan informasi tanggal yang ada di array $tgl. Dimana informasi yang ditampilkan adalah : ▪ **$hari[$tgl['wday']]** akan menampilkan informasi nama hari yang diambil dari array $hari. **$tgl['wday']** akan menghasilkan angka yang menunjuk ke urutan hari dimana 0 berarti hari Minggu. Jadi kalau **$hari [$tgl['wday']]** akan menghasilkan nama hari sesuai dengan nomor harinya. ▪ **$tgl[mday]** akan menampilkan tanggal sekarang dimulai dari 1 – 31. ▪ **$bulan[$tgl['mon']]** akan menghasilkan nama bulan sesuai dengan urutan

	bulan yang ada dalam array **$bulan**. **$tgl['mon']** akan menghasilkan nilai nomor bulan dimulai dari 1 -12. ▪ **$tgl[year]** akan menghasilkan informasi tahun sekarang.
12 – 15	Pendeklarasian fungsi **pangkat($x, $n)** yang akan menghasilkan nilai x^n. Dimana x dan n adalah parameter input.
17	Contoh pemanggilan fungsi **info_tanggal()**
18	Contoh pemanggilan fungsi **pangkat()** dengan parameter x=5 dan n=3 yang akan menghasilkan $5^3 = 125$.

Contoh lain adalah sebagai berikut :
Nama File : func2.php

```
1  <html>
2  <head>
3  <title>Function 1</title>
4  </head>
5  <body>
6  <?php
7          function ip_client(){
8                  return $_SERVER['REMOTE_ADDR'];
           }
9          function tulis_identitas($nama, $kota){
10                 echo "Nama saya $nama dari $kota";
11         }
12
13         tulis_identitas("Andri Heryandi","Sweetwater");
14         echo "<br>";
15         $ip_pengunjung=ip_client(); // simpan nilai kembalian ke variabel
16         echo "Anda memiliki IP : $ip_pengunjung"; // tampilkan isi variable
17 ?>
18 </body>
19 </html>
```

Jika dieksekusi, maka akan menghasilkan tampilan seperti di bawah ini.



Nilai Kembalian (Return Value)

Nilai kembalian adalah suatu nilai yang diberikan/dikembalikan ketika suatu function telah selesai dieksekusi. Ada dua jenis function berdasarkan nilai kembalian yaitu :

1. Function tanpa nilai kembalian
2. Function dengan nilai kembalian

Function tanpa nilai kembalian sering pula disebut sebagai procedure. Penggunaan function tanpa nilai kembalian dapat dilihat pada contoh func1.php dan func2.php yaitu pada function

info_tanggal() dan tulis_identitas(). Jika dilihat pada isi function, tidak terdapat perintah return. Jadi function tanpa nilai kembalian hanya mengerjakan perintah dari awal function ({) sampai akhir function (}).

Sedangkan untuk function dengan nilai kembalian, dapat dilihat pada file yang sama yaitu pada function ip_client() dan function pangkat(). Perbedaan besar dapat dilihat pada adanya perintah return (lihat di akhir function). Perintah return digunakan untuk mengirimkan nilai ke baris dimana function tersebut dipanggil.

Sebagai contoh

```
 1  <html>
 2  <head><title>Test Fungsi</title></head>
 3  <body>
 4  <?php
 5          function pangkat($x, $n)
 6          {
 7              return pow($x,$n);
 8          }
    ?>
 9  5<sup>3</sup> = <?php echo pangkat(5, 3);?>
10  </body>
11  </html>
```

Keterangan :

1. Garis yang solid (tidak terputus) menandakan pemanggilan function. Ketika function dipanggil dengan cara pangkat(5,3), maka alur eksekusi akan pindah ke baris di mana ada function pangkat dengan mengirimkan angka 5 sebagai parameter $x dan angkat 3 sebagai parameter $n.
2. Kemudian isi function dieksekusi. Karena hanya ada 1 perintah, maka perintah yang dieksekusi adalah perintah pow(5,3) yang akan menghasilkan perhitungan 5 pangkat 3 yaitu 125. Hasil dari pow(5,3) yaitu 125 kemudian diberikan ke perintah return. Jadi bisa dianggap bahwa pemanggilan return pow(5,3) akan menghasilkan perintah return 125.
3. Perhatikan garis yang putus-putus.Ketika perintah return dieksekusi maka nilai yang ada disebelah return yaitu 125 akan dikirim ke yang memanggilnya (baris 9).
4. Setelah function dipanggil maka perintah “echo pangkat(5,3)” akan menghasilkan perintah “echo 125”, sehingga akan menampilkan angka 125 di browser.

Hasil return dapat juga diisikan ke suatu variable terlebih dahulu, baru kemudian ditampilkan.

Variable Function

PHP memiliki suatu kemampuan yang sangat menarik yaitu mempu untuk mengeksekusi variable function. Variable function adalah sebuah pemanggilan dinamis terhadap suatu fungsi yang namanya ditentukan pada waktu eksekusi. Meskipun tidak terlalu banyak digunakan dalam kebanyakan aplikasi web, variable function dapat mengurangi ukuran kode dan kerumitan kode program, bahkan dapat menghilangkan kondisi statement yang tidak diperlukan.

Sebuah pemanggilan variable function dilakukan dengan memanggil nama variable diikuti dengan diapit tanda kurung. Dalam bagian antara tanda kurung pembuka dan penutup, boleh diisikan dengan parameter sesuai dengan pendeklarasian variablenya. Secara umum pemanggilan variable function dapat dilakukan dengan cara berikut :

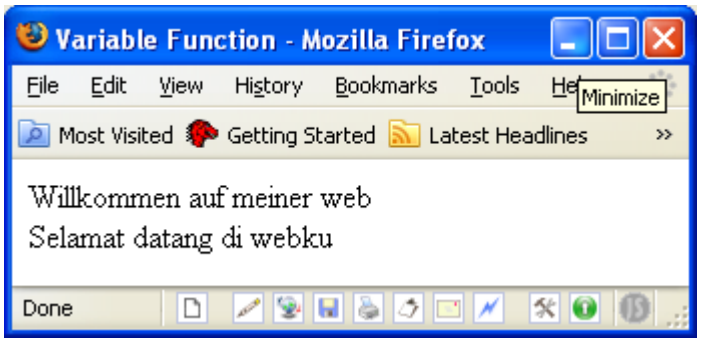
$Nama_Fungsi(daftar_parameter).

Contoh penggunaan variable function :

Nama file : var_func.php

```
1  <html>
2  <head>
3  <title>Variable Function</title>
4  </head>
5  <body>
6  <?php
7          function indonesia(){
8                  echo "Selamat datang di webku<br>";
           }
9          function inggris(){
10                 echo "Welcome to my web<br>";
11         }
12         function jerman(){
13                 echo "Willkommen auf meiner web<br>";
14         }
15         function italia(){
16                 echo "Benvenuti nel Mio Web<br>";
17         }
18         $bahasa="jerman";
19         $bahasa(); // Akan memanggil functi jerman();
21         $bahasa="indonesia";
22         $bahasa(); // akan memanggil function indonesia();
23 ?>
24 </body>
25 </html>
```

Jika dieksekusi, maka akan menghasilkan tampilan sebagai berikut :



Membuat Library Function

Library function adalah salah satu cara yang paling efisien untuk menghemat waktu ketika membangun aplikasi. Sebagai contoh, jika anda sering menggunakan sebuah fungsi dalam banyak file, maka sangatlah merepotkan jika fungsi itu harus ditulis ulang di setiap file yang memanggilnya. Hal yang lebih merepotkan lagi adalah ketika kita harus mengedit isi dari fungsi tersebut, maka kita harus mengedit sebanyak file yang memanggil fungsi tersebut. Oleh karena itu sebaiknya setiap fungsi yang sering digunakan lebih baik disimpan dalam file yang terpisah. Selain fungsi, boleh pula kita menyimpan beberapa variable atau konstanta.

Untuk memanggil sebuah file library function digunakan perintah include() atau require(). Selain include() dan require() ada juga include_once() dan require_once(). Perbedaannya adalah jika suatu include ke suatu file dilakukan selama lebih dari 1 kali dalam suatu file, maka akan

menghasilkan error karena dianggapnya ada pendeklarasian ulang (redeclare), tetapi jika menggunakan include_once() atau require_once() maka kejadian tersebut dapat dihindari.

Contoh penggunaan include() dan require() adalah sebagai berikut :

```
1  <?php
2          include("file1.php”);
3      require("file2.php”);// sama dengan pemanggilan dengan include
4      ..
5      ..
6      Pemanggilan_fungsi();
7      Pemanggilan_fungsi();
8  ?>
```

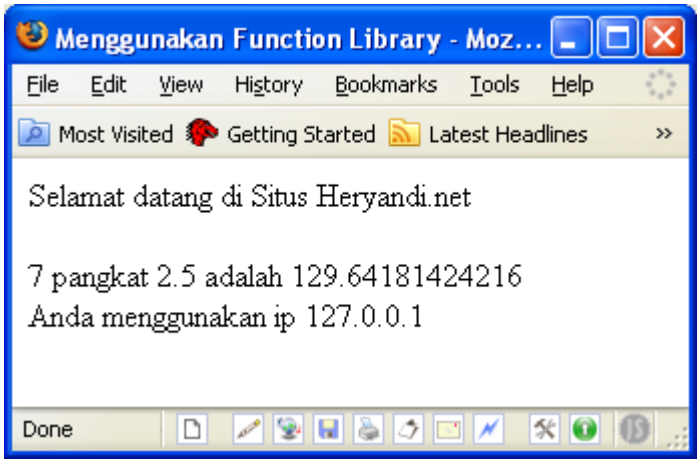
Contoh pembuatan library function dapat dilihat pada :
Nama File : my_func.php

```
1  <?php
2          $email_admin="admin@heryandi.net";
3      $nama_situs="Situs Heryandi.net";
4      function info_tanggal()
5          {
6              $hari=array("Minggu","Senin","Selasa","Rabu","Kamis","Jum'at","Sabtu");
7              $bulan=array("","Januari","Februari","Maret","April","Mei","Juni","Juli",
8                                  "Agustus","September","Oktober","November","Desember");
               $tgl=getdate();
9              echo $hari[$tgl['wday']].", $tgl[mday]-".$bulan[$tgl['mon']]."-$tgl[year]";
10         }
11         function pangkat($x, $n)
12         {
13             return pow($x,$n);
14         }
15         function ip_client(){
16                 return $_SERVER['REMOTE_ADDR'];
17         }
18         function tulis_identitas($nama, $kota){
19                 echo "Nama saya $nama dari $kota";
20         }
21 ?>
```

Contoh pemanggilan library function adalah :
Nama File : call_func.php

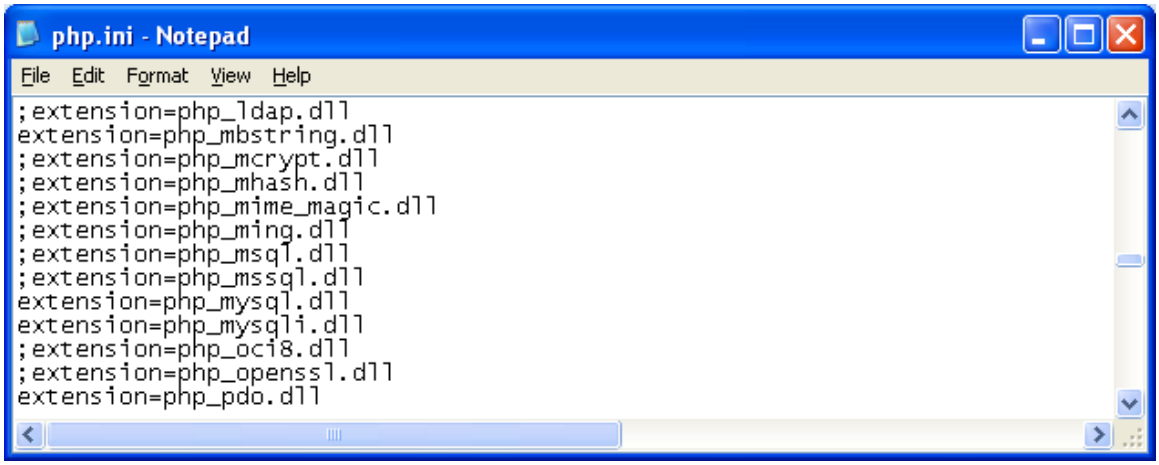
```
1  <html>
2  <head>
3  <title>Menggunakan Function Library</title>
4  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5  </head>
6  <body>
7  <?php
8          include_once("my_func.php"); // memanggil isi library my_func.php
9
10     // Memanggil variable yang dideklarasikan dalam library funcion
11         echo "Selamat datang di $nama_situs<br><br>";
12
13     // Memanggil function yang dideklarasikan dalam library function
14         echo "7 pangkat 2.5 adalah ".pangkat(7,2.5)."<br>";
15         echo "Anda menggunakan ip ".ip_client();
16 ?>
17 </body>
18 </html>
```

Jika dieksekusi, maka akan menghasilkan tampilan seperti di bawah ini.



Built-in Function (Function Internal dari PHP)

PHP menyediakan banyak fungsi yang dapat digunakan. Ada pula fungsi yang membutuhkan ekstensi PHP khusus yang jika tidak disertakan akan menghasilkan error “undefinded function” jika dipanggil. Contoh jika anda ingin PHP kita dapat berinteraksi dengan server database MySQL, maka kita harus meload ekstensi MySQL agar PHP dapat mengenalnya. Salah satu cara untuk mengaktifkan atau menonaktifkan suatu ekstensi, biasanya dilakukan dengan mengedit file PHP.ini. Lihat gambar di bawah ini

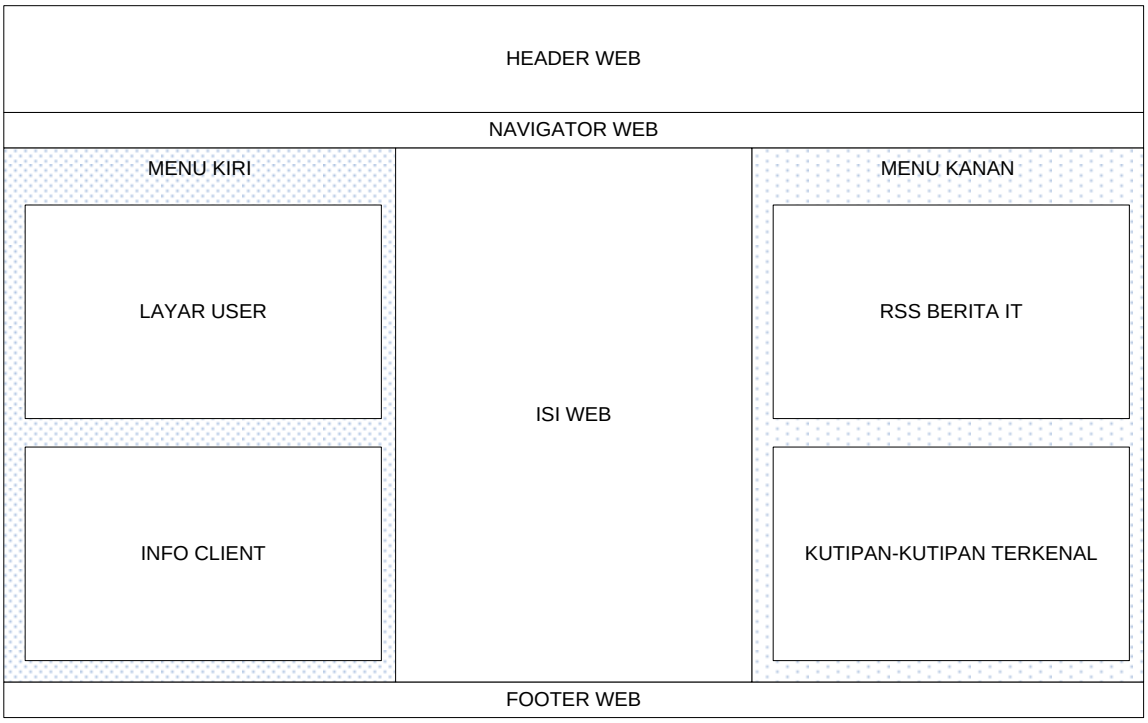


Jika anda ingin mengaktifkan suatu ekstensi, maka hilangkan tanda titik koma yang ada di awal baris. Dan jika ingin menonaktifkan ekstensi, maka cukup dengan memberi tanda titik koma di awal baris. Pada gambar di atas terlihat bahwa ekstensi untuk mengakses database server Microsoft SQL Server masih belum aktifkan, maka untuk mengaktifkannya adalah cukup menghapus tanda titik koma didepan ekstensi php_mssql.dll. Setelah itu simpan file php.ini tersebut dan restart web servernya agar dapat meload PHP dengan konfigurasi terbarunya.

Untuk mengetahui fungsi apa saja yang didukung oleh PHP, silahkan periksa manual PHP. (lihat di CD pendukung perkuliahan).

Contoh Penerapan Function

Contoh Kasus :
Anda diminta untuk membuat suatu situs yang mempunyai layout seperti di bawah ini.



Adapun halaman yang ada dalam situs tersebut adalah :

1. Halaman utama (index.php)
2. Halaman profile (profile.php)
3. Halaman download (download.php)

Setiap halaman situs mempunyai layout yang sama dengan isi yang berbeda sesuai keperluannya.

Dalam situs tersebut harus dapat menampilkan fasilitas-fasilitas berikut :

1. Menampilkan header web. Header web mungkin berupa teks, gambar atau animasi (gif, flash, applet)
2. Menampilkan navigator web yang berisi link ke halaman yang tersedia di situs
3. Menampilkan footer web. Footer web berisi copyright dari yang membuat situs.
4. Menampilkan menu sebelah kiri yang terdiri dari LAYAR USER dan INFO CLIENT
    a. LAYAR USER adalah suatu bagian yang menampilkan FORM LOGIN jika user belum login, dan akan menampikan INFO USER jika user telah login
    b. INFO CLIENT adalah suatu bagian yang menampilkan INFO CLIENT yang terdiri dari ip yang digunakan, browser yang dipakai, system operasi yang dipakai atau kalau memungkinkan adalah lokasi client.
5. Menampilkan isi web. Isi web akan berbeda untuk setiap halaman.
6. Menampilkan menu kanan yang terdiri dari RSS BERITA IT dan KUTIPAN-KUTIPAN TERKENAL.
    a. RSS BERITA IT adalah RSS (Really Simple Syndication) yang memanggil berita IT yang terbaru. Misalnya memanggil layanan RSS dari detikinet.com.
    b. KUTIPAN-KUTIPAN TERKENAL akan berisi kutipan-kutipan terkenal yang diambil dari situs lain. Kutipan ini akan berubah-ubah ketika orang lain

mengakses. Contoh kutipan-kutipan dapat menggunakan fasilitas dari **http://quotes4all.net/quotes.html.**

Berdasarkan hal di atas, maka kita memerlukan beberapa function yaitu :

1. Function header_web() untuk menampilkan header web.
2. Function navigator_web() untuk menampilkan navigator web.
3. Function footer_web() untuk menampilkan footer web.
4. Function form_login() untuk menampilkan layar login.
5. Function info_user() untuk menampilkan layar login.
6. Function layar_user() untuk menampilkan layar user.
7. Function info_client() untuk menampilkan layar info client.
8. Function menu_kiri() untuk menampilkan menu di sebelah kiri yang terdiri dari pemanggilan layar_user() dan info_client().
9. Function rss_berita_it() untuk menampilkan rss berita it dari http://detikinet.com/feed/
10. Function kutipan() untuk menampilkan kutipan-kutipan terkenal dari situs **http://quotes4all.net/quotes.html.**
11. Function menu_kanan() untuk menampilkan menu di sebelah kanan yang terdiri dari pemanggilan rss_berita_it() dan kutipan().

Untuk lebih mudah, maka buat dulu file library function yang berguna untuk menyimpan function-function yang akan digunakan dalam situs ini ke dalam file lib_func.php

Contoh isi untuk lib_func.php adalah sebagai berikut :

```
1  <?php
2          function header_web(){
3          ?>
4          <center><h1>Situs Heryandi.net</h1></center>
5          <?php
6          }
7          function footer_web(){
8          ?>
           <center>
9                  <small>Situs ini dibangun oleh<br>
10                         TZS Web Development Team
11                 </small>
12         </center>
13         <?php
14         }
15         function navigator_web(){
16                 ?>
17                 | <a href="index.php">Home</a>
18                 | <a href="profile.php">Profile</a>
19                 | <a href="download.php">Download</a> |
20                 <?php
21         }

22         function form_login(){
23         ?>
24                 <form method=post action="login.php">
25                 <table border=1 width="100%">
26                 <tr><td colspan=2 align="center">LOGIN USER</td></tr>
27                 <tr><td>Username</td><td><input type="text" name="username"
   maxlength="8" size="9"> </td></tr>
28                 <tr><td>Password</td><td><input type="password" name="userpass"
   maxlength="8" size="9"> </td></tr>
```

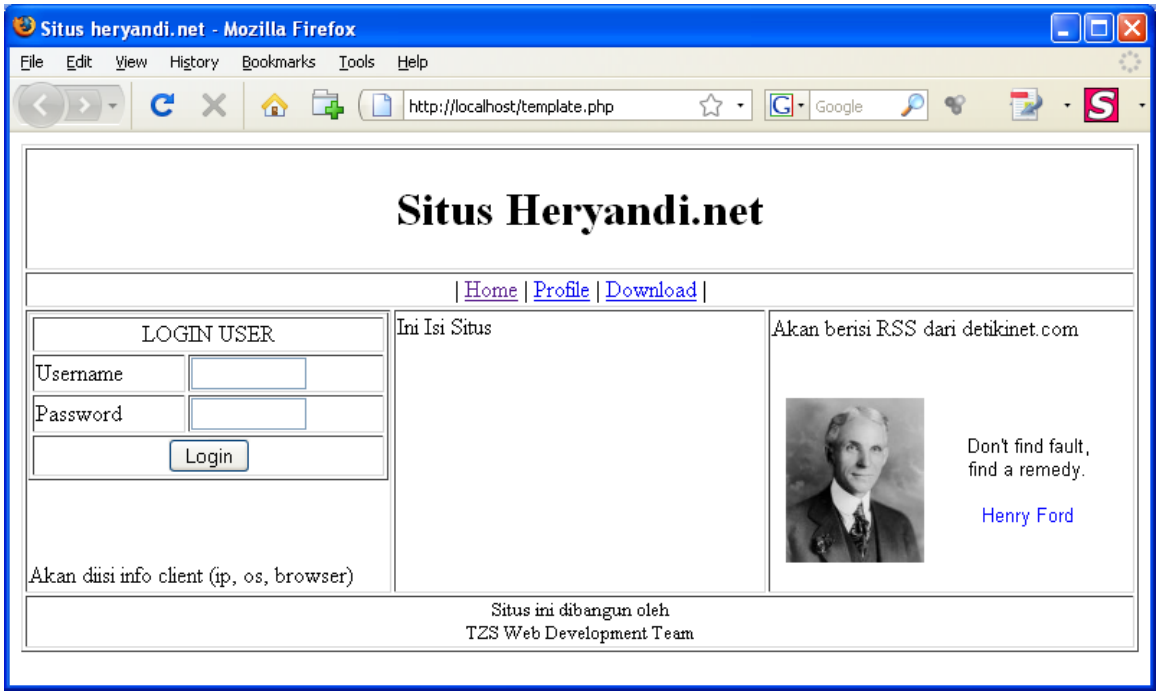
```
29                  <tr><td colspan=2 align="center"><input type="submit"
    name="btn_submit" value="Login"></td></tr>
30                  </table>
31                  </form>
32          <?php
33          }
34          function layar_user(){
35                  $telahlogin=false;// Nanti diisi perintah pemeriksaan status login
36                  if($telahlogin==false)
37                          form_login();
38                  else
39                          info_user();
40          }
41          function info_user(){
42            echo "Akan diisi info user jika telah login";
43          }
44          function info_client(){
45            echo "Akan diisi info client (ip, os, browser)";
46          }
47          function menu_kiri(){
48                  layar_user();
49                  echo "<br><br>";//pemisah
50                  info_client();
51          }
52          function rss_berita_it(){
53                  echo "Akan berisi RSS dari detikinet.com";
54          }
55          function kutipan(){
56          ?>
57                  <iframe src=" http://quotes4all.net/quotes.html" width="100%"
    frameborder="0" scrolling="no"></iframe>
58          <?php
59          }
60          function menu_kanan(){
61                  rss_berita_it();
62                  echo "<br><br>";
63                  kutipan();
64          }
65  ?>
```

Setelah library dibuat, maka buatlah layout untuk dasar (template) dari situs yang akan dibangun.
Contoh layout dasar (template) :
Namafile : template.php

```
 1  <html>
 2  <head>
 3  <?php include("lib_func.php"); ?>
 4  <title>Situs heryandi.net</title>
 5  </head>
 6  <body>
 7  <table width="100%" align="center" border=1>
 8  <tr><td colspan=3 align="center"><?php header_web();?></td></tr>
    <tr><td colspan=3 align="center"><?php navigator_web();?></td></tr>
 9  <tr>
10          <td width="250px" valign="top"><?php menu_kiri();?></td>
11          <td valign="top">Ini Isi Situs</td>
```

```
12          <td width="250px" valign="top"><?php menu_kanan();?></td>
13      </tr>
14      <tr><td colspan=3><?php footer_web();?></td></tr>
15      </table>
16      </body>
17      </html>
```

Jika dieksekusi maka perhatikan apakah layout sudah sesuai dengan yang diinginkan. Jika belum silahkan diperbaiki dulu baik tulisan atau warnanya. Jika layout telah sesuai dengan yang diinginkan maka template akan seperti gambar di bawah ini.



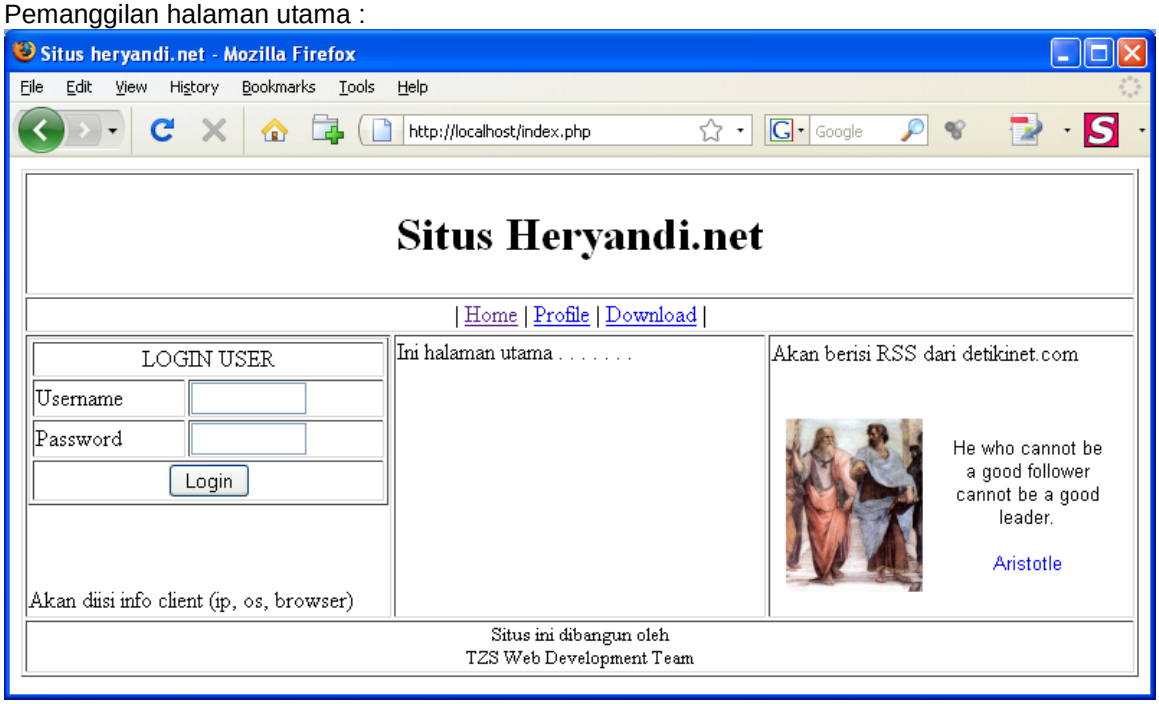
Catatan : Kutipan hanya bisa muncul kalau ditest secara online (terkoneksi internet)

Jika template telah selesai dan telah sesuai dengan keinginan, maka langkah selanjutnya adalah membuat file index.php, profile.php dan download.php dengan layout dasar yang ada di template.php. Adapun caranya adalah :

1. Copy file template.php, kemudian edit namafilenya menjadi index.php, kemudian buka editor php, dan edit pada bagian "Ini Isi Situs" sesuai dengan kegunaan halaman utama yang diinginkan.
2. Copy file template.php, kemudian edit namafilenya menjadi profile.php, kemudian buka editor php, dan edit pada bagian "Ini Isi Situs" sesuai dengan kegunaan halaman profile yang diinginkan.
3. Copy file template.php, kemudian edit namafilenya menjadi download.php, kemudian buka editor php, dan edit pada bagian "Ini Isi Situs" sesuai dengan kegunaan halaman download yang diinginkan.

Jika telah selesai maka kita mempunyai 3 file php yang mempunyai layout yang sama tetapi berbeda isi.

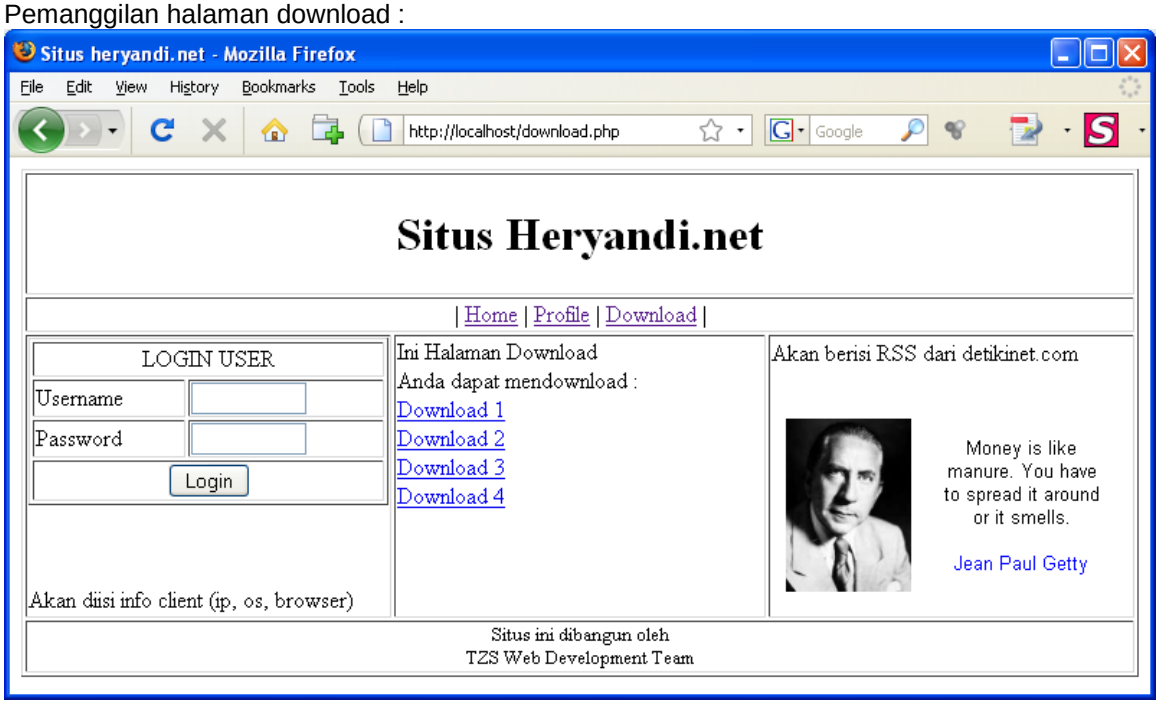
Pemanggilan halaman utama :



Pemanggilan halaman profile :



Pemanggilan halaman download :



Pemanggilan halaman download ketika variable $telahlogin diisi true (lihat bagian Layar User.

