
PANDUAN PRAKTIS

PEMROGRAMAN WEB II

Wirasto S. Karim

September 2012

Hak cipta buku ini tetap pada penulis. Tetapi buku ini bebas untuk diperbanyak, dikutip baik sebagian atau seluruhnya bahkan disebarluaskan selama bukan untuk tujuan komersial, dan dengan syarat tidak menghapus atau merubah atribut penulis.

PENGANTAR

Panduan ini berasal dari materi mengajar saya tahun lalu yang masih tercecer di beberapa berkas. Baik yang dibuat dengan *word processor*, berkas presentasi maupun dalam berkas *plain-text*. Tahun ini saya satukan dan beri embel-embel panduan praktis agar lebih mudah dibagikan ke mahasiswa, untuk mereka baca dan pelajari.

Panduan ini sebenarnya masih sangat belum selesai. Banyak bagian yang belum lengkap dan kemungkinan ada juga kalimat yang salah ketik disana-sini. Namun terpaksa saya rilis apa adanya karena sudah dibutuhkan untuk proses perkuliahan. *Insyallah* panduan ini akan terus saya perbaiki dan lengkapi agar lebih memenuhi kebutuhan perkuliahan.

Jika Anda mendapatkan panduan ini dalam bentuk tercetak (atau *fotocopy-an*), maka Anda tetap bisa mendapatkan berkas aslinya secara gratis di:

http://repository.ung.ac.id/categori/materi_kuliah/

Gorontalo, September 2012

Wirasto S. Karim

wirastokarim@gmail.com

BAB I

FORMULIR

FORMULIR

Untuk membuat halaman formulir digunakan tag form

```
<form method="post|get" action="__url__">  
</form>
```

- Atribut method ada 2 pilihan, yaitu menggunakan post atau get. Ini digunakan untuk menyatakan bagaimana masukan-masukan dalam formulir dikirim ke "action"
- Atribut action berisi url berkas yang akan memproses data masukan yang dikirimkan

Field Text

Digunakan untuk masukan teks yang tidak terlalu panjang. Field text dibuat dengan menggunakan tag input.

```
<input type="text">
```

Text Area

Digunakan untuk masukan teks yang panjang dan banyak. Text area dibuat dengan menggunakan tag textarea.

```
<textarea></textarea>
```

Radio Button

Digunakan untuk masukan berupa pilihan dimana yang dipilih hanya salah satu saja. Radio button dibuat juga dengan tag input namun dengan type radio.

```
<input type="radio"> Pilihan 1  
<input type="radio"> Pilihan 2
```

CheckBox

Digunakan untuk masukan berupa pilihan dimana yang dipilih boleh lebih dari satu. Checkbox dibuat juga dengan tag input namun dengan type checkbox.

```
<input type="checkbox"> Pilihan 1  
<input type="checkbox"> Pilihan 2  
<input type="checkbox"> Pilihan 3
```

ComboBox

Digunakan untuk masukan berupa pilihan dimana yang dipilih hanya salah satu saja dan pilihan ditampilkan dalam model drop-down list. Combobox dibuat dengan tag select dan tiap pilihan dibuat dengan tag option.

```
<select>  
    <option>Pilihan 1</option>  
    <option>Pilihan 2</option>  
    <option>Pilihan 3</option>  
</select>
```

Button

Digunakan untuk pembuatan tombol formulir. Dibuat dengan tag input dengan type yang tergantung masing-masing tombol.

```
<input type=submit>  
<input type=reset>
```

CONTOH FORMULIR

1. Formulir Standar

```
form.html  
<html>  
<head>  
    <title>Formulir</title>  
</head>  
<body>  
  
<h2>BIODATA</h2>  
  
<form method="post" action="proses.php">  
<table>  
    <tr>  
        <td>Nama</td>  
        <td>:</td>  
        <td><input type="text" name="nama"></td>
```

```

</tr>
<tr>
    <td>NIM</td>
    <td>:</td>
    <td><input type="text" name="nim"></td>
</tr>
<tr>
    <td>Jenkel</td>
    <td>:</td>
    <td>
        <input type="radio" name="jenkel" value="p"> Pria
        <input type="radio" name="jenkel" value="w"> Wanita
    </td>
</tr>
<tr>
<td>Agama</td>
    <td>:</td>
    <td>
        <select name="agama">
            <option value="islam">Islam</option>
            <option value="kristen">Kristen</option>
            <option value="yahudi">Yahudi</option>
        </select>
    </td>
</tr>
<tr>
<td>Hobi</td>
    <td>:</td>
    <td>
        <input type="checkbox" name="hobi1" value="p"> Pacaran
        <input type="checkbox" name="hobi2" value="s"> Selingkuh
        <input type="checkbox" name="hobi3" value="m"> Makan
    </td>
</tr>
<tr>
<td>Keterangan</td>
    <td>:</td>
    <td><textarea name="ket"></textarea></td>
</tr>
<tr>
    <td></td>
    <td></td>
    <td>

```

```

        <input type="submit" name="simpan" value="Simpan">
        <input type="reset" name="reset" value="Kosongkan">
    </td>
</tr>
</table>
</form>

</body>
</html>

```

2. Formulir Dengan CSS

Modifikasi script diatas (form.html) dengan menambahkan bagian ini diantara tag head, tepatnya dibawah tag title. Untuk melihat perbedaan simpan perubahannya dengan nama lain, misalnya formcss.html

formcss.html

```

<style type="text/css">
    table {
        border:1px solid #B4BBCD;
        background:#ECEFF6;
        font-size:13px;
    }

    input, select, textarea {
        border:1px solid #B4BBCD;
    }

    textarea {
        width:260px;
        height:80px;
    }

    .tombol {
        background-color: #3B5998;
        border: 1px solid #D8DFEA;
        color: #FFFFFF;
        font-size: 11px;
        padding:2 8 2 8;
    }
</style>

```

BAB II

PREDEFINED VARIABLE DAN MODULARISASI

PREDEFINED VARIABLE

Predefined variable adalah variabel yang sudah digunakan oleh php dan tidak bisa digunakan sebagai nama variabel lagi oleh programmer. Ada beberapa predefined variable, diantaranya adalah:

1. Variabel \$_GET

Variabel \$_GET berasal dari parameter url atau bisa juga berasal dari masukan lewat formulir yang dikirim dengan method get.

```
get.php
```

```
<?php
echo $_GET[nama];
?>
```

Simpan berkas diatas kedalam document root web server di localhost dan buka lewat browser dengan menuju alamat <http://localhost/get.php?nama=Wirasto>

```
formget.html
```

```
<html>
<head>
    <title>Formulir GET</title>
</head>
<body>

<form method="get" action="formget.php">
Nama : <input type="text" name="nama"> <input type="submit" value="Kirim">
</form>

</body>
</html>
```

```
formget.php
```

```
<?php
echo $_GET[nama];
?>
```

Simpan kedua berkas diatas kedalam document root web server di localhost dan buka lewat browser dengan menuju alamat <http://localhost/formget.html>

2. Variabel \$_POST

Variabel \$_POST berasal dari http post atau bisa juga berasal dari masukan lewat formulir yang dikirim dengan method post.

formpost.html

```
<html>
<head>
    <title>Formulir POST</title>
</head>
<body>

<form method="post" action="formpost.php">
Nama : <input type="text" name="nama"> <input type="submit" value="Kirim">
</form>

</body>
</html>
```

formpost.php

```
<?php
echo $_POST[nama];
?>
```

Simpan kedua berkas diatas kedalam document root web server di localhost dan buka lewat browser dengan menuju alamat <http://localhost/formpost.html>

3. Variabel \$_SESSION

Variabel \$_SESSION berasal dari sesi pemakaian maupun proses. Sebelum menggunakan variabel ini, wajib untuk menjalankan fungsi session_start terlebih dahulu. Variabel \$_SESSION inilah yang sering digunakan untuk menyimpan data sesi pemakaian dari aplikasi. Contohnya adalah digunakan untuk menyimpan data login pemakai. Yang nantinya bisa digunakan pula untuk proses validasi apa seseorang sudah melakukan proses login atau belum, menggunakan datanya dihalaman yang berbeda dan lain sebagainya.

index.php

```
<?php
session_start();
$_SESSION[nama]="Wirasto S. Karim";

echo "<a href='ceksession.php'>Cek Session</a>";
?>
```

ceksession.php

```
<?php
session_start();
echo $_SESSION[nama];
?>
```

Simpan kedua berkas diatas kedalam document root web server di localhost dan buka lewat browser dengan menuju alamat <http://localhost/>. Karena memiliki berkas utama (index.php), maka kita tidak perlu menyebutkan berkas yang akan dibuka, karena secara otomatis kalau tidak disebutkan maka berkas utamalah yang akan dibuka.

MODULARISASI

Modularisasi berarti program yang dibuat dengan php akan dibagi kedalam beberapa modul. Setiap modul atau script dikembangkan secara terpisah dengan tujuan dan fungsi khusus yang nantinya bisa digunakan kembali. Dengan begitu maka proses pengembangan aplikasi akan lebih cepat karena fungsi-fungsi atau operasi-operasi tertentu tidak perlu ditulis secara berulang-ulang ditiap script ataupun ditiap aplikasi yang berbeda.

1. Require

Jika berkas modul tidak ditemukan, maka akan menampilkan pesan Fatal Error, dan eksekusi script akan dihentikan.

kepala.php

```
<html>
<head>
    <title>Modularisasi</title>
```

```
</head>
<body>

<h2>Ini Modul kepala.php</h2>
```

kaki.php

```
<h2>Ini Modul kaki.php</h2>

</body>
</html>
```

index.php

```
<?php
require "kepala.php";
echo "<h1>Ini Halaman Utama</h1>";
require "kaki.php";
?>
```

2. Include

Jika berkas modul tidak ditemukan, maka akan menampilkan pesan Warning, dan eksekusi script tetap dilanjutkan.

kepala.php

```
<html>
<head>
    <title>Modularisasi</title>
</head>
<body>

<h2>Ini Modul kepala.php</h2>
```

kaki.php

```
<h2>Ini Modul kaki.php</h2>

</body>
</html>
```

index.php

```
<?php
include "kepala.php";
echo "<h1>Ini Halaman Utama</h1>";
include "kaki.php";
?>
```

Kedua cara diatas memiliki pasangan masing-masing yaitu `require_once` dan `include_once`. Dimana kedua cara ini bekerja sama halnya dengan pasangannya (`require` dan `include`) namun dengan tambahan sifat, kalau modul yang sudah dipanggil tidak akan dipanggil lagi.

BAB III

DASAR MySQL

APA ITU MySQL?

MySQL merupakan software yang digunakan untuk menampung dan mengelola data terstruktur yang saling berelasi. MySQL adalah Database Manajemen System (DBMS) yang paling populer dan bisa diunduh dengan cuma-cuma lewat internet di website resminya, www.mysql.com.

MySQL CLIENT

Untuk bisa mengakses database mysql yang ada di server, harus menggunakan program yang disebut dengan mysql client. MySQL client yang paling populer dan sudah menjadi bawaan dari mysql adalah mysql-client. Selain itu ada juga yang berbasis web, namun bukan bawaan dari mysql yaitu phpMyAdmin.

Untuk membuka mysql-client bawaan dari mysql, ketik mysql di terminal atau command prompt.

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 199
Server version: 5.1.62-0ubuntu0.11.10.1 (Ubuntu)

Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

MANIPULASI DATA

- Membuat Database (contoh: kuliah)

```
CREATE DATABASE kuliah;
```

- Membuka Database

```
USE kuliah;
```

- Membuat Tabel Mahasiswa

```
CREATE TABLE mhs (  
    user VARCHAR(20) PRIMARY KEY,  
    password VARCHAR(65) NOT NULL,  
    namalengkap VARCHAR(255) NOT NULL,  
    nim VARCHAR(10) NOT NULL UNIQUE  
);
```

- Memasukkan Data Baru ke dalam Tabel Mahasiswa

```
INSERT INTO mhs (user, password, namalengkap, nim) VALUES ('admin',  
MD5('admin'), 'Si Admin', '123456789');
```

```
INSERT INTO mhs (user, password, namalengkap, nim) VALUES ('wirasto',  
MD5('wirasto'), 'Wirasto Karim', '223456789');
```

```
INSERT INTO mhs (user, password, namalengkap, nim) VALUES ('shizu',  
MD5('shizu'), 'Shizuru', '323456789');
```

- Mengubah Data dalam Tabel Mahasiswa

```
UPDATE mhs SET password=MD5('kunci'), namalengkap='Gayus Tambunan' WHERE  
user='admin';
```

- Menghapus Data dalam Tabel Mahasiswa

```
DELETE from mhs WHERE user='admin';
```

- Menampilkan Data yang ada dalam Tabel Mahasiswa

```
SELECT * FROM mhs;
```

- Mengubah Engine Tabel Mahasiswa

```
ALTER TABLE mhs ENGINE=InnoDB;
```

- Membuat Tabel Mata Kuliah

```
CREATE table mk (  
    no INT PRIMARY KEY AUTO_INCREMENT,  
    kmk VARCHAR(100) NOT NULL,  
    namamk VARCHAR(255) NOT NULL,  
    user VARCHAR(20) NOT NULL,  
    UNIQUE(kmk, user),  
    FOREIGN KEY (user) REFERENCES mhs(user) ON UPDATE CASCADE ON DELETE  
CASCADE  
) ENGINE=InnoDB;
```

- Memasukkan Data Baru ke dalam Tabel Mata Kuliah

```
INSERT INTO mk (kmk, namamk, user) VALUES ('001', 'Web I', 'wirasto');
```

```
INSERT INTO mk (kmk, namamk, user) VALUES ('002', 'Web II', 'wirasto');
```

```
INSERT INTO mk (kmk, namamk, user) VALUES ('001', 'Web I', 'shizu');
```

BAB IV

PHP-MySQL

PHP-MySQL

Fungsi-fungsi php untuk berkomunikasi dengan mysql begitu banyak. Diantaranya adalah sebagai berikut:

- Membuat sambungan ke Database

```
mysql_connect("localhost", "user", "password");
```

- Membuka Database

```
mysql_select_db("kuliah");
```

- Melakukan Query

```
$hasilQuery=mysql_query("SELECT * FROM mhs");
```

- Menampilkan Data

```
$data=mysql_fetch_object($hasilQuery);  
echo $data->namalengkap;
```

PROGRAM SEDERHANA

Program ini menunjukkan cara menambah data mahasiswa dan juga menampilkan daftar mahasiswa yang ada.

```
index.php
```

```
<?php  
require "db.php";  
?>  
<html>  
<head>  
    <title>MyWeb</title>  
</head>  
<body>  
  
<table border=1>  
<tr>
```

```

        <th>No</th>
        <th>User</th>
        <th>Nama Lengkap</th>
        <th>NIM</th>
    </tr>
    <?php
    $query=mysql_query("select * from mhs");
    $i=1;
    while($data=mysql_fetch_object($query) ) {
        echo "
        <tr>
            <td>$i</td>
            <td>$data->user</td>
            <td>$data->namalengkap</td>
            <td>$data->nim</td>
        </tr> ";
        $i++;
    }
    ?>
</table>

<br>
<a href='tambahmhs.php'>Tambah</a>
</body>
</html>

```

db.php

```

<?php
mysql_connect("localhost", "root", "");
mysql_select_db("kuliah");
?>

```

tambahmhs.php

```

<?php
require "db.php";
?>
<html>
<head>
    <title>MyWeb</title>
</head>
<body>

```

```
<H1>TAMBAH MAHASISWA</H1>
```

```
<?php
```

```
if ($_POST[simpan]) {
    $user=$_POST[user];
    $pwd=$_POST[pwd];
    $nama=$_POST[nama];
    $nim=$_POST[nim];

    if ($user=="") {
        echo "Masukkan nama user!";
    } else if ($pwd=="") {
        echo "Masukkan password!";
    } else if ($nama=="") {
        echo "Masukkan nama lengkap!";
    } else if ($nim=="") {
        echo "Masukkan nim!";
    } else {
        $pwd=md5($pwd);
        $proses=mysql_query("insert into mhs (user, password, namalengkap,
nim) values ('$user', '$pwd', '$nama', '$nim')");
        if ($proses) {
            echo "Berhasil menambahkan $nama";
        } else {
            echo "Gagal menambahkan data";
        }
    }
    echo "<br><br>";
}
```

```
echo "
```

```
<form method=post action=tambahmhs.php>
```

```
<table>
```

```
<tr>
```

```
<td>User</td>
```

```
<td>:</td>
```

```
<td><input type=text name=user value='$_POST[user]'/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Password</td>
```

```
<td>:</td>
```

```
<td><input type=password name=pwd></td>
```

```
</tr>
```

```
<tr>
  <td>Nama Lengkap</td>
  <td>:</td>
  <td><input type=text name=nama value='$_POST[nama]''></td>
</tr>
<tr>
  <td>NIM</td>
  <td>:</td>
  <td><input type=text name=nim value='$_POST[nim]''></td>
</tr>
<tr>
  <td></td>
  <td></td>
  <td><input type=submit name=simpan value=Simpan></td>
</tr>
</table>
</form>
<br>

<a href='index.php'>Lihat Mahasiswa</a>";
?>

</body>
</html>
```

BAB V

OBJECT ORIENTED

OBJECT

Perbedaan utama Object Oriented Programming (OOP) atau pemrograman berbasis objek dengan pemrograman terstruktur adalah data dan kode program dalam OOP tergabung menjadi satu entitas yang disebut Objek. Tiap objek biasanya mewakili satu persoalan, yang memiliki property/atribut dan method.

CLASS

Class adalah script yang digunakan sebagai cetakan untuk membuat objek. Class mendefinisikan property yang dimiliki oleh objek serta method yang dapat dilakukan oleh objek tersebut.

Class Mahasiswa

Property :

nama

Method :

setNama(nama)

getNama()

```
class.php
```

```
<?php
class Mahasiswa {
    private $nama;

    function setNama($namaAnda)
    {
        $this->nama=$namaAnda;
    }

    function getNama()
    {
```

```

        return $this->nama;
    }
}

$mhs=new Mahasiswa();

$mhs->setNama("Nazarudin");
echo $mhs->getNama();

echo "<br>";

$mhs->setNama("Jalaludin");
echo $mhs->getNama();
?>

```

CONSTRUCTOR DAN DESTRUCTOR

- Constructor (pembangun) adalah method khusus yang tereksekusi otomatis ketika objek dari class tersebut dibuat. Konstruktor harus diberi nama `__construct()`
- Destructor (perusak) adalah method khusus yang dipanggil otomatis saat objek dimusnahkan. Destruktor harus diberi nama `__destruct()`

bagunhancur.php

```

<?php
class Mahasiswa {
    private $nama;

    function __construct()
    {
        echo "Object dibangun...<br>";
    }

    function __destruct()
    {
        echo "<br>Object dihancurkan...";
    }

    function setName($namaAnda)
    {
        $this->nama=$namaAnda;
    }
}

```

```

        function getNama()
        {
            return $this->nama;
        }
    }

    $mhs=new Mahasiswa();

    $mhs->setNama("Nazarudin");
    echo $mhs->getNama();

    echo "<br>";

    $mhs->setNama("Jalaludin");
    echo $mhs->getNama();
?>

```

INHERITANCE

Dalam inheritance (pewarisan), sebuah class turunan mewarisi class induk. Oleh karena mewarisi, maka secara teoritis semua properti dan method dari induk akan dibawa, dan menjadi bagian dari class turunan.

waris.php

```

<?php
class Mahasiswa {
    private $nama;

    function __construct()
    {
        echo "Object dibangun...<br>";
    }

    function __destruct()
    {
        echo "<br>Object dihancurkan...";
    }

    function setNama($namaAnda)
    {

```

```

        $this->nama=$namaAnda;
    }

    function getNama()
    {
        return $this->nama;
    }
}

class Kelas extends Mahasiswa {
    private $kelas;

    function setKelas($kelasAnda)
    {
        $this->kelas=$kelasAnda;
    }

    function getKelas()
    {
        return $this->kelas;
    }
}

$kelas=new Kelas();

$kelas->setNama("Nazarudin");
echo $kelas->getNama();

echo "<br>";

$kelas->setKelas("MI-01");
echo $kelas->getKelas();
?>

```

OVERRIDING

Dalam overriding, sebuah class turunan mendeklarasikan ulang method yang ada di class induk dan merubah operasi yang dilakukan.

timpa.php

```
<?php
class Mahasiswa {
    private $nama;

    function __construct()
    {
        echo "Object dibangun...<br>";
    }

    function __destruct()
    {
        echo "<br>Object dihancurkan...";
    }

    function setNama($namaAnda)
    {
        $this->nama=$namaAnda;
    }

    function getNama()
    {
        return $this->nama;
    }
}

class Kelas extends Mahasiswa {
    private $kelas;

    function setKelas($kelasAnda)
    {
        $this->kelas=$kelasAnda;
    }

    function getKelas()
    {
        return $this->kelas;
    }

    function getNama()
    {
        return "Sapa jo ngana suka";
    }
}
```

```

    }
}

$kelas=new Kelas();

$kelas->setNama("Nazarudin");
echo $kelas->getNama();

echo "<br>";

$kelas->setKelas("MI-01");
echo $kelas->getKelas();
?>

```

MySQLi

php-mysqli adalah modul objek php hasil pengembangan dari php-mysql. Mysqli dikembangkan untuk menghasilkan performansi serta fleksibilitas yang lebih baik.

mysqli.php

```

<?php
$db=new mysqli("localhost", "root", "", "kuliah");

echo "
<table border=1>
<tr>
    <th>No</th>
    <th>User</th>
    <th>Nama Lengkap</th>
    <th>NIM</th>
</tr>
";

$query=$db->query("select * from mhs");
$i=1;
while($data=$query->fetch_object()) {
    echo "
    <tr>
        <td>$i</td>
        <td>$data->user</td>
        <td>$data->namalengkap</td>
        <td>$data->nim</td>
    </tr>
";
}

```

```
        </tr>
        ";
        $i++;
    }

    echo "
</table>
";
?>
```

BAB VI

AJAX

APA ITU AJAX?

Asynchronous JavaScript And XML (AJAX) merupakan sebuah istilah untuk teknik pemanfaatan Javascript dalam mengontrol class object XMLHttpRequest untuk berkomunikasi dengan server kemudian merefresh atau mengupdate content yang ada dalam halaman web tanpa melakukan reload keseluruhan halaman.

XMLHTTPREQUEST

Kelebihan utama AJAX terletak pada pemanfaatan object XMLHttpRequest untuk berkomunikasi dengan web server secara *siluman* dalam melakukan request (permintaan).

Untuk dapat mengembangkan aplikasi web dengan menggunakan ajax ini kita perlu mengkombinasikan beberapa hal berikut:

- Javascript, untuk membuat object XMLHttpRequest yang kita gunakan untuk berkomunikasi dengan server secara *behind the scene*.
- DOM (Document Object Model), hasil proses yang diterima akan kita tampilkan dengan memanipulasi objek DOM yang telah kita persiapkan sebelumnya untuk menampilkan data hasil proses yang diberikan server.
- XML (eXtensible Markup Language), format data yang dikembalikan oleh server, data XML ini siap dibaca dan ditampilkan untuk mengupdate content pada halaman web.

PROGRAM SEDERHANA

1. Dasar Ajax

Program ini menunjukkan cara memuat data dari script lain tanpa perlu merefresh halaman yang sudah terbuka.

index.php

```
<html>
  <head>
    <title>Belajar AJAX</title>
    <script language=javascript>
      var http = false;

      if (window.XMLHttpRequest) {
        http = new XMLHttpRequest();
      } else if (window.ActiveXObject) {
        http = new ActiveXObject("Microsoft.XMLHTTP");
      }

      function muat()
      {
        http.abort();
        http.onreadystatechange=function() {
          if(http.readyState == 4) {
            document.getElementById('isi').innerHTML
= http.responseText;
          }
        }
        http.open("GET", "back.php", true);
        http.send(null);
      }
    </script>
  </head>
  <body>

  Pemanfaatan class object XMLHttpRequest untuk berkomunikasi dengan server <br>
  <input type=button value="Tampilkan" onClick="muat()">

  <div id=isi> </div>

</body>
</html>
```

back.php

```
<h2>Wow, Anda sedang belajar AJAX</h2>
```

2. Login

Program ini menunjukkan cara melakukan validasi apa user dan password yang dimasukkan untuk login sudah benar atau sebaliknya.

db.php

```
<?php
mysql_connect("localhost", "root", "");
mysql_select_db("kuliah");
?>
```

index.php

```
<?php
require_once("db.php");
?>
<html>
<head>
    <title>Login</title>
    <script language=javascript>
        var http = false;

        if (window.XMLHttpRequest) {
            http = new XMLHttpRequest();
        } else if (window.ActiveXObject) {
            http = new ActiveXObject("Microsoft.XMLHTTP");
        }

        function login()
        {
            http.abort();
            http.onreadystatechange=function() {
                if(http.readyState == 4) {
                    document.getElementById('pesan').innerHTML =
http.responseText;
                }
            }
            var user=document.getElementById('nama').value;
            var pwd=document.getElementById('sandi').value;
            http.open("GET", "proses.php?u="+user+"&p="+pwd, true);
            http.send(null);
        }
    </script>
</head>
```

```

<body>

<div id=pesan></div>
<table>
  <tr>
    <td>User</td>
    <td>:</td>
    <td><input type=text name=nama id=nama></td>
  </tr>
  <tr>
    <td>Password</td>
    <td>:</td>
    <td><input type=password name=sandi id=sandi></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td><input type=button name=kirim value='Kirim'
onClick='login();'></td>
  </tr>
</table>

</body>
</html>

```

proses.php

```

<?php
require_once("db.php");

$user=$_GET[u];
$pwd=md5($_GET[p]);

$query=mysql_query("select user from mhs where user='$user' and
password='$pwd'");
if (@mysql_num_rows($query) > 0) {
    echo "<b>Akses diterima :)</b>";
} else {
    echo "<b>Akses ditolak!</b>";
}
?>

```

3. Tambah Mahasiswa

Program ini menunjukkan cara menambah data mahasiswa dan juga menampilkan daftar mahasiswa yang ada tanpa memuat kembali semua halaman.

db.php

```
<?php
mysql_connect("localhost", "root", "");
mysql_select_db("kuliah");
?>
```

index.php

```
<html>
<head>
  <title>MAHASISWA</title>
  <script language=javascript>
    var http = false;

    if (window.XMLHttpRequest) {
      http = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
      http = new ActiveXObject("Microsoft.XMLHTTP");
    }

    function buka(file, target)
    {
      http.abort();
      http.onreadystatechange=function() {
        if(http.readyState == 4) {
          document.getElementById(target).innerHTML =
http.responseText;
        }
      }
      http.open("GET", file, true);
      http.send(null);
    }

    function kirim()
    {
      var user=document.getElementById("user").value;
      var pwd=document.getElementById("pwd").value;
      var nama=document.getElementById("nama").value;
      var nim=document.getElementById("nim").value;
```

```

        var url="proses.php?user="+user;
        url=url+"&pwd="+pwd;
        url=url+"&nama="+nama;
        url=url+"&nim="+nim;

        buka(url, "formulir");
        setTimeout(refresh, 1000);
        return false;
    }

    function refresh()
    {
        buka("daftar.php", "daftar");
        document.getElementById("formulir").innerHTML="";
    }

    buka("daftar.php", "daftar");
</script>
</head>
<body>

<input type=button value=Tambah onClick="buka('tambah.php', 'formulir')">
<input type=button value=Refresh onClick="buka('daftar.php', 'daftar')">

<br>
<br>

<div id=daftar><img src=loading.gif></div>

<br>

<div id=formulir></div>

</body>
</html>

```

proses.php

```

<?php
require_once("db.php");

$user=$_GET[user];
$pwd=md5($_GET[pwd]);

```

```

$nama=$_GET[nama];
$nim=$_GET[nim];

$query=mysql_query("insert into mhs (user, password, namalengkap, nim) values
('$user', '$pwd', '$nama', '$nim')");
if ($query) {
    echo "<i>Penambahan data berhasil</i>";
} else {
    echo "<i>Penambahan data gagal!</i>";
}
?>

```

daftar.php

```

<?php
require_once("db.php");
sleep(1);
echo "
<table border=1>
<tr>
    <th>No</th>
    <th>User</th>
    <th>Nama Lengkap</th>
    <th>NIM</th>
</tr>";

$query=mysql_query("select * from mhs");
$i=1;
while($data=mysql_fetch_object($query)) {
    echo "
    <tr>
        <td>$i</td>
        <td>$data->user</td>
        <td>$data->namalengkap</td>
        <td>$data->nim</td>
    </tr>
    ";
    $i++;
}

echo "
</table> ";
?>

```

tambah.php

```
<form method=get action='' onSubmit='return kirim() '>
<table style='border:1px solid gray;background:#D0D3CB;'>
<tr>
    <td>User</td>
    <td>:</td>
    <td><input type=text name=user id=user></td>
</tr>
<tr>
    <td>Password</td>
    <td>:</td>
    <td><input type=password name=pwd id=pwd></td>
</tr>
<tr>
    <td>Nama Lengkap</td>
    <td>:</td>
    <td><input type=text name=nama id=nama></td>
</tr>
<tr>
    <td>NIM</td>
    <td>:</td>
    <td><input type=text name=nim id=nim></td>
</tr>
<tr>
    <td></td>
    <td></td>
    <td><input type=submit name=simpan value=Simpan></td>
</tr>
</table>
</form>
```

BAB VII

WEBSERVICE

APA ITU WEBSERVICE?

WebService merupakan layanan yang memungkinkan dua buah sistem atau lebih yang masing-masing independen untuk dapat saling berkomunikasi seperti halnya client dan server. Dengan webservice seorang programmer bisa menembus batasan hardware dan software sebab tidak perlu memikirkan perbedaan diantaranya dan lebih berfokus pada solusi yang akan dikembangkan atau diberikan.

PROGRAM SEDERHANA

Program ini menunjukkan cara sederhana menggunakan teknologi webservice untuk proses login

```
index.php
<html>
    <head>
        <title>Belajar Web Service</title>
    </head>
<body>
<?php
if ($_POST[aksi]) {
    $user=$_POST[username];
    $pass=$_POST[password];

    $url="http://localhost/service.php?username=$user&password=$pass";

    $bacaxml=simplexml_load_file($url);
    foreach($bacaxml->response as $respon) {
        if ($respon=="TRUE")
            echo "Login Sukses";
        else
            echo "Login Gagal";
    }
}
?>
<h1>Form Login</h1>
```

```

<form method="post" action="index.php">
<table>
<tr>
    <td>Username</td>
    <td><input type="text" name="username"></td>
</tr>
<tr>
    <td>Password</td>
    <td><input type="password" name="password"></td>
</tr>
<tr>
    <td></td>
    <td><input type="submit" name="aksi" value="Login"></td>
</tr>
</table>
</form>

</body>
</html>

```

service.php

```

<?php
mysql_connect("localhost", "root", "");
mysql_select_db("kuliah");

$user=$_GET[username];
$pass=$_GET[password];
$query=mysql_query("select user, password from mhs where user='$user' and
password=md5('$pass')");
if (@mysql_num_rows($query)==1)
    $respon="TRUE";
else
    $respon="FALSE";

header("Content-Type: text/xml");
echo "<?xml version='1.0'?>";

echo "
<data>
<response>$respon</response>
</data>";
?>

```

PUSTAKA

MySQL Documentation Team, *MySQL Manual*, <http://dev.mysql.com/doc/>, 2012

PHP Documentation Group, *PHP Manual*, <http://php.net>, 2012

Prasetyo Didik Dwi, *Solusi Menjadi Web Master melalui Manajemen Web dengan PHP*, Elex Media Komputindo, Bandung 2005