

# Kumpulan Tutorial AJAX – PHP

## Table of Contents

1.	Tips Optimasi skrip PHP().....	4
2.	Pengenalan Codeigniter.....	6
3.	Belajar jQuery mudah .....	9
4.	Coding Best Practise.....	14
5.	Create, Update dan Delete Data Master Dengan Ajax.....	17
6.	Tips Merancang User Interface Form Pencarian Data.....	24
7.	Membuat Menu Tree Dengan PHP dan JQuery.....	28
8.	Skrip/Hal yang sering dibuat/digunakan oleh web-developer().....	31
9.	Belajar Ajax dengan prototype.js.....	32
10.	Submit Form Dengan Ajax Menggunakan jQuery.....	34
11.	AJAX – Asynchronous JavaScript And XML (PART 1) .....	36
12.	Tutorial AJAX, teknik membuat aplikasi web lebih interactive.....	40
13.	Membuat File pdf dengan PHP .....	44
14.	Membuat URL dengan Alamat PHP Menjadi HTML .....	45
15.	Membuat chart dengan Codeigniter dan php libchart.....	47
16.	Script PHP untuk Membuat Dokumen RTF .....	49
17.	Tips Membuat Web Full Ajax Sederhana .....	53
18.	Membuat PHP .....	56
19.	Membuat pilihan Tanggal dengan ComboBox pada PHP .....	59
20.	Source Upload dan View Gambar dengan PHP.....	61
21.	File Upload Progress dengan PHP 5 dan APC.....	63
22.	Membuat Slideshow Headline Berita Sederhana Dengan JQuery (+Efek Animasi) .....	71
23.	Polling Dengan Ajax Dan Menampilkan Hasil Polling Dengan Grafik Pie / Oval.....	75
24.	Membuat Thumbnail Image Dengan PHP dan Image Slideshow Dengan Thickbox JQuery .....	79
25.	Membuat Data Pagging Dengan Ajax .....	85
26.	Membuat Tab Pada Halaman Web Dengan Lebih Mudah.....	88
27.	Modernisasi Proses Input Form Anda Dengan JQuery Ajax Form.....	92
28.	Mematikan Interface Dan Menampilkan Alert Dengan Pilihan Design Sendiri.....	95

29. Membuat Web Service dengan PHP .....	98
30. Directory Trees With PHP And jQuery .....	103
31. Create a Photo Admin Site Using PHP and jQuery .....	108
32. PHP ajax login form using Jquery .....	122
33. HOWTO: PHP and jQuery upload progress bar .....	125
34. How to Validate Forms in both sides using PHP and jQuery .....	128
35. Auto-Complete Field with jQuery, JSON & PHP .....	140
36. Create a shoutbox using PHP and AJAX (with jQuery) .....	142
37. Tutorial - CakePHP Ajax "Quick Save" with jQuery .....	153
38. PHP & jQuery image upload and crop v1.2 .....	155
39. FlickrScrolr Explained: PHP .....	159
40. PHP Login script .....	168
41. Tips Membuat Script PHP Pengolah Password dengan MD5 .....	174
42. Tentang Letak Script Pengolah Form .....	177
43. Membuat Tampilan Data Terurut Berdasarkan Kolom Tabel .....	181
44. Teknik Membuat On Page Paging .....	185
45. Membuat Advanced Paging dengan PHP .....	190
46. Input Data Ke MySQL Secara Simultan Via Form dengan PHP .....	200
47. Script PHP Updating Data Secara Simultan (Studi Kasus Update Nilai Kuliah Mahasiswa) .....	204
48. Membuat Script PHP untuk Hapus Data dengan Multi Selection .....	209
49. Membuat Script Grafik PHP Untuk Menampilkan Indeks Prestasi Mahasiswa (Studi Kasus) .....	215
50. Beberapa Function PHP Untuk Keperluan File System .....	221
51. Teknik Scripting Mencegah SQL Injection di PHP .....	225
52. Mengenkripsi Parameter GET Method Untuk Keamanan .....	228
53. Membuat Sistem Login Aplikasi Multi User .....	231
54. Script PHP untuk Menampilkan Siswa Belum Bayar SPP (A Study Case) .....	236
55. Membuat Script Random Image dengan PHP .....	239
56. Apa Sih Gunanya Session di PHP? .....	242
57. Script PHP untuk Membuat Dokumen RTF .....	248
58. Menguak Rahasia Penggunaan Cookies di Bisnis Afiliasi .....	252
59. Menu Horizontal Multi Level Menggunakan jQuery & PHP .....	255
60. jQuery Mega Menu .....	258

61. Export hasil query MySql di web ke Excel .....	261
62. Menampilkan Format Angka ke Rupiah di PHP .....	263
63. Shopping cart sederhana.....	265
64. Creating a simple PHP forum tutorial .....	271
65. Membuat Shopping Cart Sederhana Membuat Shopping Cart Sederhana Menggunakan PHP ....	285
66. PHP/MySQL Tutorial (Add, View, Edit & Delete from to MySQL database) Live Example (News System ).....	289
67. Creating A Menu System .....	305
68. Membuat Format Rupiah di PHP .....	308
69. sd .....	309
70. s .....	310

## 1. Tips Optimasi skrip PHP()

Ingin meningkatkan performa situs anda?

Banyak hal optimasi yang bisa dilakukan, beberapa diantaranya disampaikan disini. Berikut adalah beberapa method yang diharapkan bisa membantu meningkatkan performa aplikasi PHP rekan-rekan:

### 1) object code caching

Ada beberapa paket object-code-caching yang tersedia di Internet (ada yang free, ada yang komersial):

- Ioncube: <http://ioncube.com/>
- Zend Encoder: [http://zend.com/products/zend\\_safeguard](http://zend.com/products/zend_safeguard)
- Turck MMCache: <http://freshmeat.net/projects/turck-mmcache/>

### 2) Template systems

Template-system berbeda dengan caching.

Template-system sangat berguna pada kondisi dimana data-statik dari halaman anda tidak harus di-reload.

Template-system php yang sering banyak digunakan (free):

- [Smarty Templates](#)
- [Pear Templates](#)
- [PHP savant](#)

### 3) Distributed object caching systems

Yang sering digunakan adalah [memcached](#).

*This type of system makes your overall site faster by caching the majority of your database data into a large memory pool..*

*memcached is a high-performance, distributed memory object caching system, generic in nature, but intended for use in speeding up dynamic web applications by alleviating database load.*

### 4) PHP variables that can be set

variables\_order = 'GPC'

register\_argc\_argv = 'Off'

register\_globals = 'Off' (this is a good idea to keep off for security purposes as well)

always\_populate\_raw\_post\_data = 'Off'

magic\_quotes\_gpc = 'Off'

Disable Error Logging.

Selama masih dalam masa-develop aplikasi, abaikan hal ini.

menonaktifkan logging akan meningkatkan performance cukup luar biasa.

Gunakan IP address untuk melakukan akses ke database, hindari penggunaan hostname.  
walaupun kadang kala tidak memungkinkan.

Dengan menggunakan ip, akan lebih mempercepat proses lookup.

## 5) Output Compression

Saat ini, sebagian besar browser telah mendukung sesuatu yang dinamakan *gzip-compression*. Gzip-compression dapat mengurangi ukuran output handler. Katanya, walaupun ukuran menurun hingga 80%, tp cpu-usage naik hingga 10%. yahh... ini resikonya, memang cpu jadi kerja berat, tp load-page jadi lebih singkat.

Pengaktifan dilakukan dari konfigurasi php (php.ini)

```
zlib.output_compression = On  
zlib.output_compression_level = (level) ( 1-9. cari nilai yang terbaik untuk server anda).
```

Jika menggunakan apache, bisa juga dengan memasang module **mod\_gzip**. Ada beberapa hal pembahasan yang berbeda mengenai hal ini.

## 6) Other things that may help

Pada saat menggunakan database, cukup ambil data yang memang benar-2 diperlukan. misal:

hindari penggunaan "select \* from namatable"  
tapi cukup gunakan "select namafield, namafieldlain from namatabel"

Trik yang lain juga bisa ditemui di:

<http://ilia.ws/archives/12-PHP-Optimization-Tricks.html>  
<http://www zend com/zend/trick/trick-optimizing-php.php>

## 2. Pengenalan Codeigniter



CodeIgniter adalah sebuah framework php yang dapat mempercepat developer untuk membuat sebuah aplikasi web. Ada banyak library dan helper yang berguna didalamnya dan tentunya mempermudah proses development.

### Apa Keunggulan Codeigniter?

CodeIgniter sangat ringan, terstruktur, mudah dipelajari, dokumentasi lengkap dan support yang luar biasa dari forum codeigniter. Alasan lainnya adalah

- Menggunakan pattern MVC yang mempermudah dan meningkatkan fleksibilitas
- Menghasilkan url yang search engine friendly
- Mudah untuk membuat library dan helpernya
- Jalan di PHP 4 (4.3.2+) dan 5
- Support banyak database MySQL (4.1+), MySQLi, MS SQL, Postgres, Oracle, SQLite, dan ODBC.
- Sekuritas aplikasi terjaga
- Banyak library dan helper yang mempermudah pekerjaan

### Apa itu MVC?

MVC adalah kepanjangan dari **Model View Controller**. MVC sebenarnya adalah sebuah pattern yang bertujuan memisahkan bisnis logic, data logic dan presentation logic (interface) atau simplicity memisahkan antara desain, data dan proses.

- Model: Model berhubungan dengan data dan interaksi ke database.
- View: View berhubungan dengan segala sesuatu yang akan ditampilkan ke end user. bisa itu halaman web, rss, javascript dan lain-lain
- Controller: Controller bertindak sebagai penghubung data dan view

Berikut ini adalah contoh penggunaan codeigniter untuk menampilkan berita yang ada di database.

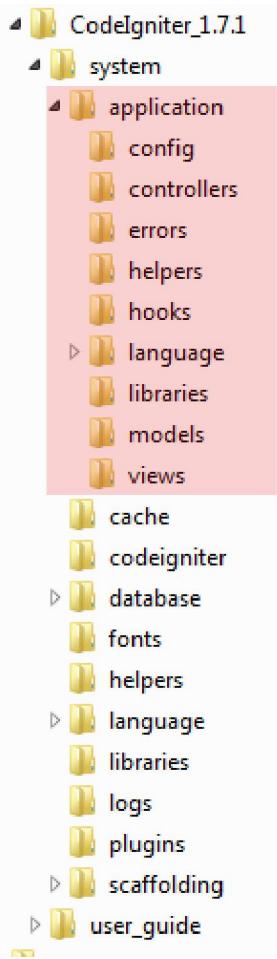
### **Langkah 1: Download Codeigniter**

Anda dapat mendownload codeigniter di [website Codeigniter](#)

### **Langkah 2: Ekstract Codeigniter**

Setelah anda meng ekstrak, letakkan di folder htdoc anda. anda akan mendapatkan struktur direktori seperti berikut ini.

Kita hanya akan bermain di folder application, di situ lah aplikasi kita diletakkan.



- Folder **system** menyimpan semua file, baik itu file aplikasi yang kita buat maupun core frameworknya.
  - Folder **application**: disinilah aplikasi yang akan kita bangun diletakkan
    - Folder **config** : tempat menyimpan semua file konfigurasi yang ada di aplikasi mulai dari database, router dan autoload dari aplikasi
    - Folder **controllers** : tempat menyimpan semua file controller.
    - Folder **errors** : tempat menyimpan semua template error aplikasi
    - Folder **helpers** : tempat menyimpan helper-helper yang bukan berasal dari CI
    - Folder **hooks** : tempat menyimpan hook yang digunakan untuk mengubah alur fungsi dari core CI
    - Folder **language** tempat menyimpan bahasa-bahasa yang akan di gunakan
    - Folder **libraries** tempat menyimpan semua library buatan kita sendiri
    - Folder **models** tempat menyimpan semua model.
    - Folder **views** tempat menyimpan semua file view aplikasi.
  - Folder **cache** tempat menyimpan semua cache yang dibuat caching library.
  - Folder **codeigniter** tempat menyimpan semua file internals CI.
- Folder **database** tempat menyimpan semua driver database drivers dan class yang akan digunakan.
- Folder **fonts** tempat menyimpan semua fonts yang digunakan image manipulation library.
- Folder **helpers** tempat menyimpan semua helper core CI.
- Folder **language** tempat menyimpan semua langguage core CI.
- Folder **libaries** tempat menyimpan semua library core CI
- Folder **logs** tempat menyimpan semua logs generated oleh CI.
- Folder **plugin** tempat menyimpan semua plugin core CI.
- Folder **scaffolding** tempat menyimpan semua file yang berfungsi sebagai scaffolding .
- Folder **user\_guide** berisi userguide/manual penggunaan CI.
- File **index.php** file yang akan menghandle semua request yang dilakukan oleh client.

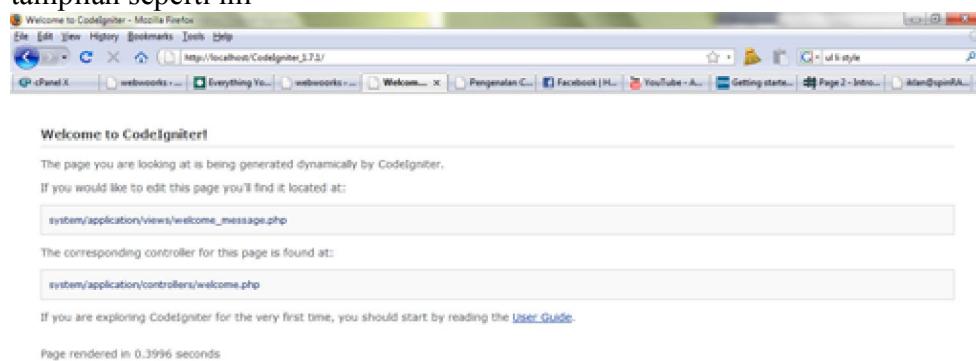
### Langkah 3: Konfigurasi Codeigniter

untuk menggunakan codeigniter setidaknya anda harus mengubah konfigurasi base\_url. Semua request dan helper yang berhubungan dengan link akan menggunakan itu. buka file open up system/application/config/config.php

```
10 $config['base_url'] = "http://localhost/belajar_ci/";
```

### Langkah 4: Testing Default Codeigniter

Mengecek apakah dengan konfigurasi dasar sudah jalan apa belum. apabila sudah jalan akan ada tampilan seperti ini



### 3. Belajar jQuery mudah



JQuery adalah sebuah framework/library JavaScript yang dapat membantu kita mempermudah dan mempercepat pengolahan DOM pada halaman web. Dengan jQuery kita dapat membuat web lebih menarik

dan interaktif dengan mudah. jQuery sudah mengautomasi pekerjaan-pekerjaan yang umum dan mempersimple code yang kompleks. Library ini sangat kecil dan mempunyai banyak plugin yang dapat mempermudah kita.

Adapun fitur-fitur yang ditawarkan oleh jQuery adalah

- Mempermudah akses dan manipulasi ke bagian page tertentu. jQuery menawarkan sebuah **selector** yang robust dan efisien untuk mengambil bagian tertentu pada dokumen yang selanjutnya bisa dimanipulasi.
- Mempermudah perubahan tampilan dokumen. jQuery dapat mengubah tampilan CSS dengan mudah.
- Merespon interaksi user dengan webpage. jQuery mempunyai cara yang sangat-sangat elegan untuk memasukkan sebuah even ke dalam salah satu bagian dari webpage.
- Menambah animasi. Kita dapat memberi animasi pada webpage kita dengan jQuery
- Mempermudah AJAX.
- 

[TOP](#)

Baiklah mari kita mulai dengan installasi jQuery. pada tutorial ini akan dibagi menjadi lima bagian yaitu instalasi, manipulasi, event, animasi dan ajax. pada artikel ini akan di bahas mengenai instalasi dan manipulasi.

#### INSTALASI JQUERY

Pertama-tama download dulu jQuery dari websitenya <http://jquery.com/>. Ada dua pilihan download. Production dan development, perbedaannya adalah production digunakan di realsite sedangkan development untuk masa developmen. Versi production sudah di kompresi sehingga kode nya sudah tidak bisa dibaca lagi dan ukurannya jauh lebih kecil. Versi developmen berguna jika kita mau mendebug jQuery.

Setelah mendownload maka load-lah jquery melalui tag script (perhatikan baris ke-7). sesuaikan attribut srcnya dengan path jquery anda.

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6 <title>My First JQuery</title>
```

```
7 <script type="text/javascript" src="jquery.js"></script>
</head>
```

### JQUERY SELECTOR

Selector adalah fungsi utama pada jQuery. semua fungsi lainnya di jQuery dapat diakses melalui selector. Penggunaan paling dasarnya adalah mempassingkan sebuah ekspressi (biasanya ekspressi css) yang kemudian selanjutnya jQuery akan mencari elemen yang cocok. pada intinya ini adalah fungsi untuk memilih elemen-elemen pad halaman web.

fungsi selector adalah `$('ekspressi')`. untuk penggunaan `$()` yang aman sebaiknya dilakukan setelah semua page DOM selesai terbaca semua. jadi webpagennya sudah lengkap baru script di eksekusi. kita dapat melakukannya dengan pada fungsi `$(document).ready()`.

Adapun parameter yang bisa digunakan pada jquery adalah Id element, class element dan element itu sendiri. Selector juga bisa mengenali hirarki sebagai parameter inputnya dan bisa memfilternya langsung dari ekspressi.

\*CATATAN : demo menggunakan layanan jsbin. anda bisa mengubah script html dan javascriptnya langsung, dengan mendouble click. pilih tab output untuk melihat hasilnya

contoh 1. mengubah semua div yang mempunyai p

demo : <http://jsbin.com/ofaku>

```
1  (silakan di utak-atik demonya)
2  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
3      "http://www.w3.org/TR/html4/loose.dtd">
4  <html>
5  <head>
6  <base href="http://docs.jquery.com" />
7  <!-- Disini kita mengeload library Jquery -->
8  <script src="http://code.jquery.com/jquery-1.3.js"></script>
9
10 <script>
11 $(document).ready(function() {
12     // memanggil semua div yang berisi p lalu ubah bordernya
13     $("div > p").css("border", "1px solid gray");
14 });
15 </script>
16 </head>
17 <body>
18 <p>Ini adalah paragraf pertama</p>
19 <div><p>Ini adalah paragraf kedua didalam div </p></div>
20
21 <p>Ini adalah paragraf Ketiga</p>
22
23 </body>
24 </html>
```

contoh 2. Memasukkan tag html kedalam elemen

Demo : <http://jsbin.com/ijopu>

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
2     "http://www.w3.org/TR/html4/loose.dtd">
3 <html>
4 <head>
5 <base href="http://docs.jquery.com" />
6 <!-- Disini kita mengeloa library Jquery -->
7 <script src="http://code.jquery.com/jquery-1.3.js"></script>
8
9 <script>
10 $(document).ready(function() {
11     //memasukkan html kedalam elemen
12     $("<div><p>Saya adalah Paragraf Tabmahan</p></div>").appendTo("body")
13     // memanggil semua div yang berisi p lalu ubah bordernya
14     $("div > p").css("border", "1px solid gray");
15
16 });
17 </script>
18 </head>
19 <body>
20 <p>Ini adalah paragraf pertama</p>
21 <div><p>Ini adalah paragraf kedua didalam div </p></div>
22
23 <p>Ini adalah paragraf Ketiga</p>
24
25 </body>
26 </html>

```

## **UJI COBA**

dari contoh diatas. coba kita buka halaman

lalu masukkan tambahkan kode ini satu persatu. lihat effect yang terjadi lalu pelajari fokuslah pada parameter yang ada pada fungsi **\$(PARAMETER)** bukan fungsi dibelakangnya karena itu akan di jelaskan setelah ini.

```

1 //contoh selector menggunakan nama tag
2 $("div").css("border", "3px solid red");
3 $("p").css("border", "3px solid blue");
4 //contoh selector menggunakan nama object/element
5 $("#id_pertama").css("color", "red");
6 //contoh selector menggunakan nama Kelas object
7 $(".merah").css("color", "red");

```

\*untuk dapat melihat perubahan script di coba satu-satu jangan langsung semua

## **JQUERY MANIPULATION**

Jquery Manipulation maksudnya penggunaan jquery untuk memanipulasi struktur dokumen. ada 7 kategori manipulasi yang disupport oleh jquery yaitu

Perubahan Content. Kita dapat merubah konten dari sebuah object dengan dua fungsi yaitu html dan text. bedanya jika menggunakan text maka semua tag-tag html akan dituliskan sebagai mana mestinya. contoh (demo <http://jsbin.com/ifoho>) :

```
1  $("p").click(function () {  
2      var htmlStr = $(this).html();  
3      $(this).text(htmlStr);  
4  });
```

Insert kedalam object. Setelah kita memilih sebuah object, kita dapat memasukkan tag html atau object kedalamnya. ada dua jenis insert yaitu append dan prepend. append menambahkan objek baru setelah value dari object tersebut, sedangkan prepend didepannya. contoh

```
1 $("p").append("<strong>Hello</strong>");  
2 $("span").appendTo("#conto"); // Objek dengan ID conto akan dimasukkan  
3           kedalam span  
4 $("p").prepend("<b>Hello </b>");  
5 $("span").prependTo("#conto");
```

Insert dikeluar object. Object yang akan ada tambahkan akan ada diluar masing-masing tag yang dipilih. contoh

```
1 $("p").before("<strong>Hello</strong>");  
2 $("p").after("<strong>Hello</strong>");  
3 $("span").insertAfter("#conto");  
4 $("span").beforeAfter("#conto");
```

Insert kesekitar object.

Mereplace object. Kita dapat mengganti isi dari object dengan syntax ini. contoh

```
1 $("button").click(function () {  
2     $(this).replaceWith("<div>" + $(this).text() + "</div>");  
3  });  
4  
5 </div>
```

Menghapus object. kita dapat menhapus object tertentu. contoh

```
1 $("button").click(function () {  
2     $("p").empty();  
3  });  
4  
5 $("button").click(function () {  
6     $("p").remove();  
7  });  
8 </div>
```

Mengopi object. Mengopi object ke tempat lain . contoh

```
1 $("b").clone().prependTo("p");  
2 <
```

## 4. Coding Best Practise

Coding Best Practise adalah cara-cara coding yang efektif dan telah teruji oleh para-para ahli dan praktisi koding. Jika ingin mendapatkan hasil code yang baik maka kita sebaiknya mengikuti standar-standar berikut ini

### 1. Coding standar

Coding standar artinya kita harus mempunyai ketetapan-ketetapan dalam menulis code. Misal penamaanan kelas, fungsi, spacing, tab, dll. Tujuannya adalah membuat kode kita gampang dibaca dan gampang di maintain. Ada beberapa coding standar yang dapat kita gunakan diantaranya:

- Pear PHP Coding Standar <http://pear.php.net/manual/en/standards.php>
- Zend Coding Standar <http://framework.zend.com/manual/en/coding-standard.html>

Contoh Kode yang buruk

```
1 <?php
2 //fungsi 1
3 if ($syarat){
4     fungsil($param_a);
5     if($a>4 && $r<4 || $p=3)
6         x('param_lagi');
7     }
8 ?>
1 <?php
2 //mengambil nilai siswa
3 if ($is_siswa)
4 {
5     display_siswa_info($id_siswa);
6     if( ($a>4) && ($r<4) || ($p=3))
7         simpan_data_siswa($data_siswa);
8 }
9
10 ?>
```

### 2. Code Commenting

Code commenting adalah sebuah aktifitas memberikan komentar pada kode yang telah kita buat. manfaat yang diperoleh adalah mempermudah kita dalam membaca kode, bisa berfungsi sebagai dokumentasi.

Contoh komentar yang mengikuti standar phpdocumentor

```
1 /**
2  * Keterangan tentang kelas
3  *
4  * @category Nama kategori
```

```

5  * @package      Nama Paket
6  * @author       pembuat code <author@example.com>
7  * @copyright    1997-2005 The PHP Group
8  * @license      http://www.php.net/license/3_0.txt  PHP License 3.0
9  * @version      Release: @package_version@
10 */
11 class Foo_Baz
12 {
13     /**
14      * keterangan tentang variabel
15      *
16      * @var string
17      */
18     public $foo = 'unknown';
19
20    /**
21     * keterangan singkat tentang fungsi
22     *
23     * @param string $arg1 argumen pertama, bertipe string
24     * @param int    $arg2 argumen kedua bertipe integer.
25     * @return int   Hasil kembalian fungsi ini
26     *
27     */
28     public function fooBar($arg1, $arg2='')
29     {
30         // code...
31         return 1;
32     }
33 }

```

### 3. Code Reuse

Code reuse ini bermaksud untuk menggunakan kembali kode-kode yang sudah pernah kita buat di project sebelumnya. Kadang-kadang kita harus melakukan refactoring pada kode yang telah kita buat. gunakanlah konsep DRY (dont repeat yourself).

### 4. Code Testing

ketika kita membuat code. biasakan untuk mengetes kode tersebut. kita juga dapat menggunakan unit test untuk menguji kode kita ini.

### 5. Source Control

Source control dapat berfungsi sebagai personal undo, tool kolaborasi kode jika kita bekerja di team. Inti penggunaan source control adalah melakukan tracking terhadap kode yang telah kita buat

### 6. Code Quality Improvement

code yang kita buat harus

- simple : Solusi yang ditawarkan harus menggunakan algoritma yang paling simple (KISS - Keep it simple stupid)

- menyelesaikan masalah : percuma jika kode yang kita buat tidak menyelesaikan masalah
- secure : kode yang kita buat harus aman
- maintainable : kode yang kita buat harus dapat di maintain/ di ubah dengan mudah
- scalable : kode yang kita buat harus scalable, tidak peduli jumlah data yang banyak, performa tetap stabil

## 5. Create, Update dan Delete Data Master Dengan Ajax

The screenshot shows a web application titled "Ajax CRUD" by Chandra Jatnika. At the top, there is a table with four rows of employee data:

No	NIP	Nama	Department	Action
1	P01	Chandra Jatnika	Programmer	Edit   Delete
2	P02	Cruzi Oldolin	Designer	Edit   Delete
3	P03	Suke Ponesix	Director	Edit   Delete
4	P04	Domisini Postinour	Manager	Edit   Delete

Below the table, a modal dialog is open with the title "Tambah Data Pegawai". It contains fields for NIP, Name, Address, Department (with a dropdown menu set to "Programmer"), and buttons for Simpan (Save) and Batal (Cancel). There is also a "close or Esc Key" link in the top right corner of the modal.

Proses **Create, Update dan Delete (CRUD)** adalah proses dasar dalam management data. Kali ini saya akan membangun proses CRUD dengan metode Ajax.

Data yang diinput adalah data karyawan, pertama-tama ciptakan dulu table karyawan dan master departement:

```
CREATE TABLE `master_pegawai` (
  `ID` INT( 5 ) NOT NULL AUTO_INCREMENT PRIMARY KEY ,
  `NIP` CHAR( 5 ) NOT NULL ,
  `Name` VARCHAR( 50 ) NOT NULL ,
  `Address` VARCHAR( 255 ) NOT NULL ,
  `Department` INT( 5 ) NOT NULL ,
  UNIQUE (`NIP`)
);
CREATE TABLE IF NOT EXISTS `master_department` (
  `ID` int(5) NOT NULL AUTO_INCREMENT,
  `DepartmentName` varchar(50) NOT NULL,
  PRIMARY KEY (`ID`)
);
```

Isi data pada master department terlebih dahulu karena pada artikel kali ini saya akan berkonsentrasi pada management data master pegawai. Library javascript yang akan digunakan adalah:

1. **JQuery**
2. **JQuery Ajax Form**, untuk memproses data dari form dan request ke ajax lebih cepat dan mudah.
3. **JQuery Thickbox**, untuk mengeluarkan popup form dengan metode Ajax.

Ada tiga file utama yang dapat anda temukan dengan fungsi masing-masing:

1. **index.php**, menampung daftar lengkap pegawai dan menyediakan link untuk tambah, edit dan hapus data pegawai. Pada **index.php** juga di deklarasikan fungsi untuk request Ajax ketika tombol submit pada form ditekan.
2. **form.php**, menampilkan form tambah atau edit data. File ini diakses melalui metode Ajax dengan bantuan popup **Thickbox**, jika file ini dipanggil dengan parameter ID pegawai maka form ini siap untuk mengedit satu record pegawai di database.
3. **proses.php**, adalah file yang menjadi tempat request Ajax untuk menambah dan mengedit data. File ini akan mengembalikan nilai **JSON** untuk kemudian di proses **index.php**.

#### ISI FILE “INDEX.PHP” PADA BAGIAN HEAD:

```
<script language="javascript" src="js/jquery.js"></script>
<script language="javascript" src="js/thickbox.js"></script>
<script language="javascript" src="js/jquery.form.js"></script>
<script language="javascript" src="js/jsCalendar.js"></script>
<script language="javascript">
    /* Fungsi addRow untuk menambahkan baris pada table pegawai setelah
       Ajax berhasil melakukan proses tambah data pegawai */
    function addRow(ID,NIP,Name,DeptName) {
        var str = '<tr id="tr'+ID+'">';
        /* cara mengetahui jumlah baris pada table pegawai saat ini
           untuk membuat nomor urut data pada table */
        var countLine = Object($('#tableBody tr')).length;
        str += '<td align="center">'+(countLine+1)+'</td>';
        str += '<td align="center">'+NIP+'</td>';
        str += '<td>'+Name+'</td>';
        str += '<td>'+DeptName+'</td>';
        str += '<td align=center><a href="form.php?width=370&height=230&id_pegawai=' + ID +
        ' class="thickbox">Edit</a> ';
        str += '| <a href="javascript:deleteRow(\'' + ID + '\')">Delete</a></td>';
        str += '</tr>';

        // kode html yang sudah disusun ditambahkan (append) ke dalam tag
        tbody
        $('#tableBody').append(str);

        /* setiap link dengan class='thickbox' pada baris yang baru dibuat
        akan
```

```

        mempunyai fungsi mengeluarkan dialog thickbox ketika
melakukan edit data */
    tb_init('#tr'+ID+' td a.thickbox');
}
// untuk mengedit data pada baris tertentu di table daftar pegawai
function editRow(ID,NIP,Name,DeptName){
    var element = '#tr'+ID; // ambil ID dari baris yang ingin diubah
    $(element+' td:nth-child(2)').text(NIP); // kolom kedua (NIP) dari
baris diubah datanya
    $(element+' td:nth-child(3)').text(Name); // kolom ketiga (Nama)
dari baris diubah datanya
    $(element+' td:nth-child(4)').text(DeptName); // kolom keempat
(Department) dari baris diubah datanya
}

// untuk menghapus baris atau data pada table pegawai
function deleteRow(ID){ // ID dari record pegawai dikirimkan sebagai
parameter
$.post(
    'delete.php',
    {id_pegawai: ID},
    function(response){
        if(response == 'ok') // jika respon dari delete.php
adalah 'ok'
            $('#tr'+ID).remove(); // hapus 1 baris
        else
            alert('Delete gagal');
    }
);
/*
fungsi submitForm akan dipanggil pada saat melakukan inputan pada
form tambah atau edit pegawai. Di dalam fungsi ini ada fungsi
ajaxSubmit
yang merupakan plugin dari jquery yaitu jquery.form.js
*/
function submitForm(){
    // beri tanda bahwa data sedang di proses dengan efek loading
fade in
    $('#divResult').text('loading...').fadeIn();
    $('#formData').ajaxSubmit({
        success: function(response){
            // sembunyikan element tempat hasil respon sementara
agar efek fade in lebih nyata
            $('#divResult').hide();
            /* respon yang diterima berupa json dengan struktur
{status:(nomor status), IDPegawai: (id dari pegawai)}*/
            if(response.status == 1){
                /* jika nilai status adalah 1 berarti ada error dan
response.text yang merupakan pesan error akan
ditampilkan di
                    element dengan ID = divResult */
                // pesan error diberi efek fade in dengan background
warna merah
            }
        }
    });
}
$( '#divResult' ).text(response.text).css({ 'color': '#FFFFFF', 'background-
color': '#FF0000' }).fadeIn();
} else if(response.status == 2){

```

```

        /* jika nilai status adalah 2 maka tandanya sedang
berlangsung
           proses penambahan data pegawai baru oleh karena
itu fungsi yang
           dipakai selanjutnya adalah addRow untuk
menambah baris di table pegawai */
           // pesan sukses diberi efek fade in dengan background
warna kuning
$('#divResult').text('Data berhasil
ditambah').css({'color':'#000000','background-color':'#FFFF00'}).fadeIn();
addRow(
    response.IDPegawai,
    $('#input[@name=NIP]').val(),
    $('#input[@name=Name]').val(),
    $('#select[@name=Department]
option:selected').text()
);
tb_remove();
else if(response.status == 3){
/* jika nilai status adalah 3 maka sedang berlangsung
proses edit data pegawai maka fungsi yang
dipanggil selanjutnya
           adalah fungsi editRow untuk mengedit baris data
pegawai yang
           sudah diubah di form */
// pesan sukses diberi efek fade in dengan background
warna kuning
$('#divResult').text('Data berhasil
diedit').css({'color':'#000000','background-color':'#FFFF00'}).fadeIn();
editRow(
    $('#input[@name=ID]').val(),
    $('#input[@name=NIP]').val(),
    $('#input[@name=Name]').val(),
    $('#select[@name=Department]
option:selected').text()
);
tb_remove();
},
dataType: 'json' // menandakan bahwa ajax menginginkan
respon berupa json
);
return false;
}
</script>

```

#### ISI FILE “INDEX.PHP” PADA BAGIAN BODY:

```

<table cellspacing="0" cellpadding="4" border="1" width="500">
<thead>
<tr bgcolor="#CCCCCC">
<th>No</th>
<th>NIP</th>
<th>Nama</th>
<th>Department</th>

```

```

        <th>Action</th>
    </tr>
</thead>
<tbody id="tableBody"><?
    include 'connect.inc.php';
    $query = mysql_query('SELECT p.*,d.DepartmentName FROM
master_pegawai p JOIN master_department d ON p.Department = d.ID');
    if($query && mysql_num_rows($query) > 0){
        $x = 1;
        while($row = mysql_fetch_object($query)){
            echo "<tr id='tr{$row->ID}'>
                    <td align='center'>$x</td>
                    <td align='center'>{$row->NIP}</td>
                    <td>{$row->Name}</td>
                    <td>{$row->DepartmentName}</td>
                    <td align='center'><a class='thickbox'
href='form.php?id_pegawai={$row->ID}&width=370&height=230'>Edit</a>
                            | <a
href='javascript:deleteRow(\"{$row->ID}\")'>Delete</a></td>
                    </tr>";
            $x++;
        }
    }
?></tbody>
</table>
<br /><a href="form.php?width=370&height=230" class="thickbox" title="Tambah
Data Pegawai">Tambah Data Pegawai</a>
```

**ISI FILE “FORM.PHP” BERIKUT PENJELASANNYA:**

```

<?
    include 'connect.inc.php';
    $id = isset($_GET['id_pegawai']) ? intval($_GET['id_pegawai']) :
false;

    if($id){
        $query = mysql_query('SELECT * FROM master_pegawai WHERE ID =
"'.$id.'"' );
        if($query && mysql_num_rows($query) == 1){
            $data = mysql_fetch_object($query);
        }else{
            die('Data pegawai tidak ditemukan');
        }
    }
?>
<div id="divResult" style="font-size:11px;text-
align:center;display:none"></div>
<form action="process.php" method="post" id="formData" onSubmit="return
submitForm();">
<table>
    <tr>
        <td>NIP</td>
        <td>:</td>
        <td><input type="text" name="NIP" size="6" maxlength="6"
value=<?=@$data->NIP?>" onKeyUp="this.value =
String(this.value).toUpperCase()" /></td>
```

```

</tr>
<tr>
    <td>Nama Pegawai</td>
    <td>:</td>
    <td><input type="text" name="Name" size="30" value="<?=@$data->Name?>" /></td>
</tr>
<tr valign="top">
    <td>Alamat</td>
    <td>:</td>
    <td><textarea name="Address" rows="6" cols="30"><?=@$data->Address?></textarea></td>
</tr>
<tr>
    <td>Departement</td>
    <td>:</td>
    <td>
        <select name="Department">
            <?
                $query = mysql_query('SELECT * FROM master_department');
                if($query && mysql_num_rows($query) > 0) {
                    while($row = mysql_fetch_object($query)) {
                        echo '<option value="'.$row->ID.'"' ;
                        if($row->ID == @$data->Department) echo ' selected';
                        echo '>'.$row->DepartmentName.'</option>';
                    }
                }
            ?>
        </select>
    </td>
</tr>
<tr>
    <td>&ampnbsp</td>
    <td>&ampnbsp</td>
    <td>
        <input type="hidden" name="ID" value="<?=@$data->ID?>" />
        <input type="submit" name="submit" value="Simpan" />
        <input type="button" value="Batal" onClick="tb_remove()" />
    </td>
</tr>
</table>
</form>

```

#### **ISI FILE “PROSES.PHP” BERIKUT PENJELASANNYA:**

```

<?
/* proses.php akan menjadi file yang di request oleh Ajax
   dan akan mengembalikan nilai dengan struktur JSON */
if(isset($_POST['ID'])){
    include 'connect.inc.php';
$ID = getPost('ID');
$NIP = getPost('NIP');
$Name = getPost('Name');
$Address = getPost('Address');

```

```

$Department = getPost('Department');
if($NIP && $Name) {
    if($ID) {
        $SQLUpdate = "UPDATE master_pegawai SET
                      NIP = '$NIP',
                      Name = '$Name',
                      Address = '$Address',
                      Department = '$Department'
                     WHERE ID = '$ID'";
        $query = mysql_query($SQLUpdate) or die(mysql_error());
        echo '{status:3}'; // memberikan respon nilai status = 3
        ketika berhasil mengedit
    } else{
        $SQLInsert = "INSERT INTO master_pegawai
                      (NIP,Name,Address,Department)
VALUES ('$NIP', '$Name', '$Address', '$Department')";
        $query = mysql_query($SQLInsert) or die(mysql_error());
        $lastID = mysql_insert_id();
        /* memberikan respon nilai status = 2 dan ID dari record
        pegawai
           ketika berhasil menambah data pegawai baru */
        echo '{status:2,IDPegawai:'.$lastID.'}';
    }
    // mengembalikan respon nilai status = 1 dan text error message
    }else echo '{status:1,text:"Lengkapi Isi Form. NIP dan Nama Harus
Diinput"}';
}
?>

```

Untuk melihat demo klik [link ini](#).

Untuk download source program klik [link ini](#).

Terima kasih atas kunjungannya, maaf sudah lama menunggu perkembangan artikel di website ini, penulis lagi banyak banget kerjaan...hehehe ^\_^

## 6. Tips Merancang User Interface Form Pencarian Data

Pada saat merancang sebuah form anda perlu memikirkan masalah user interface agar para pengguna aplikasi anda dapat dengan mudah mengoperasikan sistem yang anda buat dan mencegah adanya human error. Design user interface pada aplikasi harus diperhatikan pada setiap field yang berhubungan dengan data dari database.

Pada artikel kali ini saya ingin memberikan tips bagaimana mensiasati perancangan form pada saat pencarian data pada suatu table yang record-nya sudah banyak sehingga tidak mungkin element SELECT/Dropdown bisa menampungnya. Berikut adalah screenshootnya:

The screenshot shows a modal dialog box titled "Pilih Pegawai" with the identifier "P123". At the top right is a "close" button with the text "close or Esc Key". The dialog contains a table with four columns: "No. Peg", "Nama", "Umur", and "Alamat". The table has 17 rows, each representing an employee record. The data is as follows:

No. Peg	Nama	Umur	Alamat
P001	Abai Sudami	22	Ketapang
P002	Asun Monjoli	22	Jakarta Timur
P003	Ambili Godon	21	Kreo
P004	Bimil Pulanta	21	Kota
P005	Beni Simantupil	25	Kreo
P006	Caca Donima	21	Tangerang
P007	Cintu Ontom	23	Pamulang
P008	Cici Tomoli	24	Tangerang
P009	Coki Komplos	19	Senen
P010	Dani As burep	26	Bandung
P011	Dani S. Polcin	22	Meruya
P012	Deni Ambara	21	Kembangan
P013	Desi Kolipum	23	Petukangan
P014	Desica Aniom	26	Pancoran
P015	Desica Zukardi	26	Tebet
P016	Deri Zaki	27	Pulau Seribu
P017	Erna Sibantak	28	Blok M

Pada tampilan diatas ada satu inputan kode pegawai dan di sebelahnya adalah nama pegawai sesuai dari kode yang dimasukkan. Apabila user hafal dengan kode pegawai maka kode dapat langsung diinput dan ketika cursor sudah keluar dari inputan kode maka dengan metode Ajax, pencarian nama pegawai dengan kode yang diinput sebelumnya pun akan dilakukan. Namun apabila user lupa akan kode pegawai, dia dapat menekan tombol cari dan keluar dialog seperti gambar diatas. Dialog diatas bertujuan untuk mensupport user dalam melakukan pencarian pegawai yang ingin diinput.

Lain halnya apabila user salah menginput manual kode pegawai, maka akan ada peringatan seperti gambar dibawah. Peringatan ini sangat berguna dalam mengurangi kesalahan input user.



Untuk membuat program ini, siapkan beberapa file PHP yang nantinya akan berfungsi untuk menampilkan daftar pegawai dan pencarian pegawai. Pertama siapkan file dengan nama “daftar\_pegawai.php“:

```
<?
include 'koneksi.php';
$query = mysql_query("SELECT * FROM pegawai");
?>


| No. Peg | Nama | Umur | Alamat |
|---------|------|------|--------|
|         |      |      |        |


```

Lalu siapkan file dengan nama “cara\_pegawai.php“:

```
<?
// pastikan file ini diakses dengan mengirimkan variable POST dengan nama
// "nip"
isset($_GET['nip']) or die('Kirim parameter NIP');

include 'koneksi.php';
$nip = $_GET['nip'];
$query = mysql_query("SELECT nama FROM pegawai WHERE NIP = '$nip'");
// cara mengambil nilai nama pegawai dari query diatas
@list($nama_pegawai) = mysql_fetch_row($query);
echo $nama_pegawai; // beri response nilai nama pegawai
?>
```

Untuk formnya siapkan script seperti ini:

```

<form action="" method="post">
    Pilih Pegawai: <input type="text" name="pegawai" size="4" />
45    <!-- semua element anchor dengan class="thickbox" akan otomatis
46    menampilkan dialog thickbox,
47        isi href bisa dimasukan setting thickbox berupa width dan height
48        dari dialog thickbox -->
49    <a href="daftar_pegawai.php?width=300&height=400" class="thickbox"></a>
51    <input type="text" name="nama_pegawai" size="20" readonly />
52    <small id="divAlertPegawai"></small>
</form>

```

Untuk support javascript dalam program ini contoh scriptnya seperti ini:

```

/* fungsi selectPegawai tidak digunakan di file ini melainkan digunakan di
file daftar_pegawai
yang nantinya akan dipanggil oleh thickbox, fungsinya untuk
memasukan nilai hasil klik dari
salah satu daftar pegawai di table lalu dimasukan ke masing-masing
inputan form */
8 function selectPegawai(nip,nama){
11     $('input[@name=pegawai]').val(nip);
12     $('input[@name=nama_pegawai]').val(nama);
13     tb_remove(); // hilangkan dialog thickbox
14 }
15 $(document).ready(function(){
16     $('input[@name=pegawai]').blur( // beri event pada saat onBlur inputan
17     kode pegawai
18         function(){
19             var vNIP = $(this).val();
20             $.get('cari_pegawai.php',{nip: vNIP},
21                 function(nama_pegawai){
22                     // jika response tidak kosong nilainya maka masukkan nilai
23                     ke inputan nama pegawai
24                     if(nama_pegawai.length > 0){
25                         $('input[@name=nama_pegawai]').val(nama_pegawai);
26                     }else $('#divAlertPegawai').text('Pegawai dengan NIP
27 "'+vNIP+' Tidak Ditemukan').css('color','red');
28                     }
29                 );
30             }
31         );
32     // beri event pada saat keyup kode pegawai agar kode yang dimasukan font-
33 nya UPPERCASE semua (optional)
34     $('input[@name=pegawai]').keyup(
35         function(){
36             $(this).val(String($(this).val()).toUpperCase());
37         }
38     );
39 });

```

Sebagai catatan, saya menggunakan **JQuery + Plugin Thickbox** untuk membuat dialog thickbox dan penerapan Ajax.

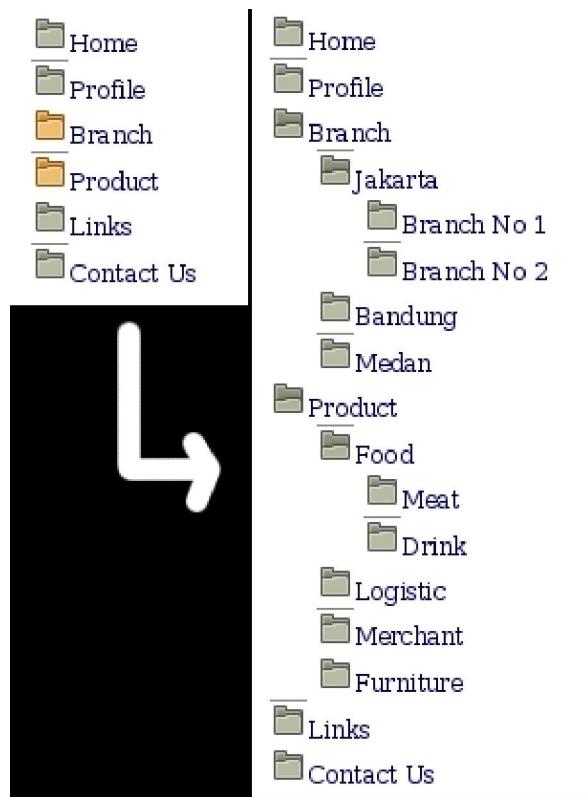
Untuk sample data pegawai download disini → [\*\*sample data pegawai\*\*](#)

Untuk demo program lihat di [\*\*link ini\*\*](#)

Untuk download program lengkap di [\*\*link ini\*\*](#)

Semoga dapat menambah wawasan... ^\_^

## 7. Membuat Menu Tree Dengan PHP dan JQuery



Pada setiap aplikasi, menu merupakan element paling penting yang berfungsi sebagai navigasi dari halaman satu ke halaman lainnya, dari module satu ke module lainnya. Pada aplikasi yang lebih professional, menu sangat diperhatikan karena merupakan sarana untuk pembagian module per masing-masing user dan group pengguna aplikasi.

Menu sebagai dasar navigasi dari suatu aplikasi haruslah dimanis atau bisa diedit, ditambah dan di custom oleh admin. Untuk itu menu setiap link-nya harus disimpan di database, berikut sample table menu yang sederhana.

```
CREATE TABLE IF NOT EXISTS `menu` (
  `id` int(5) NOT NULL AUTO_INCREMENT,
  `parent_id` int(5) NOT NULL,
  `name` varchar(50) NOT NULL,
  `url` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
);
```

Dan ini adalah contoh sample datanya.

ID	PARENT	NAME	URL
1	0	Home	?page=home
2	0	Profile	?page=profile
3	0	Branch	#
4	3	Jakarta	?page=branch&id=jakarta
5	3	Bandung	?page=branch&id=bandung
6	3	Medan	?page=branch&id=medan
7	0	Product	
8	7	Food	?page=product&id=food
9	7	Logistic	?page=product&id=logistic
10	7	Merchant	?page=product&id=merchant
11	7	Furniture	?page=product&id=furniture
12	0	Links	?page=links
13	0	Contact Us	?page=contact
14	8	Meat	?page=product&id=food&c=meat
15	8	Drink	?page=product&id=food&c=drink
16	4	Branch No 1	?page=branch&id=jakarta&c=1
17	4	Branch No 2	?page=branch&id=jakarta&c=2

Sample data untuk menu ini bisa anda download di file ini → [sample\\_menu.sql](#)

Teknik load data dari database untuk kemudian ditampilkan dalam bentuk menu berjenjang dengan format tree (pohon) adalah teknik rekursif yang dapat anda lihat di contoh source code berikut.

```

37 <?
38 /* fungsi ini akan terus di looping secara rekursif agar dapat menampilkan
39 menu dengan format tree (pohon)
40 * dengan kedalaman jenjang yang tidak terbatas */
41 function loop($data,$parent){
42     if(isset($data[$parent])){ // jika ada anak dari menu maka tampilkan
43         /* setiap menu ditampilkan dengan tag <ul> dan apabila nilai
44 $parent bukan 0 maka sembunyikan element
45         * karena bukan merupakan menu utama melainkan sub menu */
46         $str = '<ul parent="'. $parent .'>';
47         style="display:'. ($parent>0?'none':'').'">';
48         foreach($data[$parent] as $value){
49             /* variable $child akan bernilai sebuah string apabila ada sub
50 menu dari masing-masing menu utama
51             * dan akan bernilai negatif apabila tidak ada sub menu */
52             $child = loop($data,$value->id);
53             $str .= '<li>';
54             /* beri tanda sebuah folder dengan warna yang mencolok apabila
55 terdapat sub menu di bawah menu utama
56             * dan beri juga event javascript untuk membuka sub menu di
57 dalamnya */
58             $str .= ($child) ? '<a href="javascript:openTree('. $value-
59 >id.')">id.'" border="0"></a>' :
60             '';
61             $str .= '<a href="'. $value->url.'">' . $value->name . '</a></li>';
62             if($child) $str .= $child;
63         }
64         $str .= '</ul>';

```

```

65         return $str;
66     }else return false;
67 }
68 mysql_connect('localhost','root','123456');
69 mysql_select_db('test');
70
// tampilkan menu di sortir berdasar id dan parent_id agar menu ditampilkan
// dengan rapih
$query = mysql_query('SELECT * FROM menu ORDER BY id,parent_id');
$data = array();
while($row = mysql_fetch_object($query)){
    $data[$row->parent_id][] = $row; // simpan data dari database ke dalam
// variable array 3 dimensi di PHP
}
echo loop($data,0); // lakukan looping menu utama
?>

```

Dan cuma menggunakan javascript sederhana untuk membuat menu ini dapat dibuka dan ditutup contoh gambar paling atas.

```

function openTree(id){
    // ambil semua tag <ul> yang mengandung attribut parent = id dari
21 link yang dipilih
22     var elm = $('ul[@parent='+id+']');
23     if(elm != undefined){ // jika element ditemukan
24         if(elm.css('display') == 'none'){ // jika element dalam keadaan
25             tidak ditampilkan
26             elm.show(); // tampilkan element
27             $('#img'+id).attr('src','folderopen.jpg'); // ubah gambar
28 menjadi gambar folder sedang terbuka
29         }else{
30             elm.hide(); // sembunyikan element
31             $('#img'+id).attr('src','folderclose2.jpg'); // ubah gambar
32 menjadi gambar folder sedang tertutup
33         }
    }
}

```

Lihat demo selengkapnya di [Link ini](#)

Download keseluruhan program di [Link ini](#)

Semoga bermanfaat... ^\_^

## 8. Skrip/Hal yang sering dibuat/digunakan oleh web-developer()

Hampir seluruh developer php selalu menggunakan skrip/pustaka/kode berikut.

Tidak semua, tetapi sepertinya hampir semuanya memang menggunakan.

### [x] Kirim Email

fungsi mail() hampir dipastikan selalu digunakan.

Mulai dari sekedar form-kontact, reminder, notifikasi, dll

secara pribadi, lebih dianjurkan untuk menggunakan [phpmailer](#)

### [x] Manipulasi User

Pasti juga khan?

setiap cms pasti menggunakan fitur ini: login, logout, registrasi, profile, permission, etc  
pustaka yang bisa digunakan : [phpuserclass](#)

### [x] RSS Feeds

Ingin materi anda dilihat banyak orang dengan berbagi content.

Dengan menggunakan rss fees saya pikir merupakan langkah yang tepat.

### [x] Grabbing Remote Content

Jika sebelumnya untuk orang yang ingin berbagi content,

nahh.. kali ini yang dipakai oleh orang yang ingin meminta bagian tadi :)

### [x] Trackback

Bagi yang sering bermain blogging, istilah ini pasti sudah dikenal

pustaka yang bisa digunakan : [phptrackback](#)

### [x] Template

Membuat layout di dalam code pasti sangat menjemukan,

tetapi... tetap juga masih ada developer yang bermain tampilan langsung embed di dalam kode yang dibuat.

capek deh.

Nah... gunakan template.

Tata letak tampilan bisa diatur tanpa menyentuh kode langsung,

tinggal kemudian perlu beberapa baris kode untuk penyesuaian.

pustaka yang bisa digunakan: [smarty](#)

## 9. Belajar Ajax dengan prototype.js

Ya ya, saya tahu ini basi, tapi paling tidak ini adalah pengalaman saya secara pribadi yang baru tahu bentuk makhluk yang namanya AJAX. Tapi bukan Ajax Andsterdam loh, saya nggak suka sama Ajax. Sukanya ama AC Milan.

Ajax singkatan dari “**Asynchronous Javascript and XML**“. Adalah sebuah teknologi yang memungkinkan kita bekerja di web secara realtime. Yap, itulah pengertian Ajax menurut saya (maklum bahasa Inggris lemah). Dan sebelumnya saya melihat sebuah script yang membingungkan. Maklum saya itu orangnya kepengen cepet dan nggak pengen yang lambat.

Dalam hal ini masalah coding ajax. Terus terang saya sendiri orangnya aplikatif. Tidak faham dengan script yang panjang. Dan mending diberitahukan cara tercepat untuk memulai dan menggunakannya. Dan et voila, hal itu terjawab sudah dengan menggunakan prototype.js. Script ini bisa anda ambil di wordpress. Saya sengaja nggak memberikan link download-annya, karena anda bisa download wordpress sendiri. 😊

Petunjuk dan manualnya bisa dilihat [di sini](#). Saya mulai dengan membuat sebuah halaman agar tampak terlihat realtime. Karena tempat saya bekerja ini adalah distributor pulsa, maka mau tak mau kita harus tahu antrian yang saat ini ada. Entah itu tidak diproses ataupun sedang diproses oleh server.

Kasusnya seperti ini. Misalnya saat ini ada berapa antrian di server?  
Script sederhananya adalah sebagai berikut:

### lihatjmlantrian.php

```
<?php
//connection
$query = "select count(*) as jml from purchase where status < ".$_GET['st'];
$rs = mysql_query($query);
$rjml = mysql_fetch_object($rs);
$jml = $rjml->jml;
mysql_free_result;
echo $jml;
?>
```

Cukup dengan kode di atas kita sudah bisa tahu berapa jumlah antrian yang belum terproses di dalam database. Misalnya status antrian itu adalah 0 untuk menunggu dan 1 untuk sedang diproses. Dan 2-6 untuk status yang lain. Maka kita hanya akan mendapatkan data antrian yang menunggu dan sedang diproses.

Kemudian kita membuat script untuk memanggilnya. Sangat mudah.

### panggil.php

```
<?php
//header, connection, function
?>
```

```
<script type='text/javascript' src="js/prototype.js"></script>
<h2>Jml Antrian</h2><span id="jmlantrian">(loading....)</span>
<?php
//footer
?>
<script language='javascript'>
new Ajax.PeriodicalUpdater(
'jmlantrian','lihatjmlantrian.php?st=2',
{
method:'get',frequency:5,decay:2
}
);
</script>
```

Ada beberapa fungsi pada Ajax, yaitu Ajax.Request yaitu tujuan murni dari ajax, yang berupa requester. Kemudian ada Ajax.Responder, Ajax.Updater, Ajax.Option, dan ada Ajax.PeriodicalUpdater. Dan kali ini saya menggunakan Ajax.PeriodicalUpdater yang berguna untuk menampilkan refresh halaman secara terus menerus. Dan ini berguna bagi kasus seperti diatas. Yah...semuanya bisa di download di situs yang sudah saya sebutkan tadi. Mulai dari prototype-nya, API-nya, sekaligus documentasinya. Tinggalkan koding yang berbelit-belit, saatnya menggunakan resource yang sudah ada.

## 10. Submit Form Dengan Ajax Menggunakan jQuery

Kalau sebelumnya saya sudah menulis tentang cara menampilkan halaman menggunakan Ajax (menggunakan fungsi `load()` pada jQuery), maka kali ini adalah contoh penggunaan Ajax untuk mengirimkan data dalam sebuah form dan memprosesnya. Salah satu kelebihan menggunakan form berbasis Ajax dibandingkan form konvensional adalah: kita tidak perlu meninggalkan form selama form dikirimkan/diproses.

Untuk contoh ini, saya menggunakan dua buah file, yang pertama adalah file `ajaxform.html` untuk menampilkan form nya, dan file `proses.php` untuk memproses data yang dikirimkan dan menampilkan hasilnya.

Script pada file `ajaxform.html` nya adalah sbb:

```
1 $(document).ready(function() {
2
3     $('#loading').show();
4     $('#result').hide();
5 ).ajaxStart(function() {
6     $('#loading').show();
7     $('#result').fadeIn('slow');
8 });
9
10
11     $('#myForm').submit(function() {
12         $.ajax({
13             type: 'POST',
14             url: $(this).attr('action'),
15             data: $(this).serialize(),
16             success: function(data) {
17                 $('#result').html(data);
18             }
19         })
20         return false;
21     });
22 })
```

Untuk mengimplementasikan Ajax, pada file `form.html` ini saya menggunakan fungsi `ajax()` pada jQuery. Fungsi ini memiliki sebuah argumen yaitu berupa object (pasangan key/value), dan yang akan saya gunakan di antaranya sbb:

- `type`: jenis request yang dipakai, bisa 'POST' atau 'GET'
- `url`: url yang akan digunakan untuk memproses data. karena pada form sudah terdapat nilai `action` (`proses.php`) maka saya tinggal mengambil nilai dari `action` tsb menggunakan `$(this).attr('action')`
- `data`: data yang dikirimkan, dalam format querystring. untuk menghasilkan querystring dari form, saya menggunakan fungsi `serialize()`

- success: fungsi yang akan dijalankan jika request berhasil, dengan sebuah argumen berupa data yang dikembalikan dari server, dalam hal ini adalah hasil output dari file proses.php (hasil output ini akan saya tampilkan ke dalam sebuah div dengan id="result" )

Sedangkan file proses.php yang akan memproses data yang dikirimkan, isinya adalah sbb:

```

1 <?php
2 //validasi
3 if (trim($_POST['nim']) == '') {
4     $error[] = '- NIM harus diisi';
5 }
6 if (trim($_POST['nama']) == '') {
7     $error[] = '- Nama harus diisi';
8 }
9 if (trim($_POST['tempat_lahir']) == '') {
10    $error[] = '- Tempat Lahir harus diisi';
11 }
12 //dan seterusnya
13
14 if ($error) {
15     echo '<b>Error</b>: <br />' . implode('<br />', $error);
16 } else {
17     /*
18      jika data mau dimasukkan ke database,
19      maka perintah SQL INSERT bisa ditulis di sini
20     */
21
22     echo '<b>Form berhasil disubmit. Berikut ini data anda:</b>';
23     echo '<br />';
24     foreach ($_POST as $k => $v) {
25         $data .= "$k : $v<br />";
26     }
27     echo $data;
28 }
29 die();
30 ?>
```

Setelah file proses.php ini berhasil dijalankan maka hasil outputnya akan ditampilkan pada element div yang berada pada file ajaxform.html

Untuk lebih jelasnya silakan lihat demo:

[Demo Ajax Form](#)

Dan anda bisa mendownload source code nya di sini:



## 11. AJAX – Asynchronous JavaScript And XML (PART 1)

Istilah AJAX technology belakangan ini sering terdengar dalam pengembangan aplikasi web, berikut sedikit gambaran tentang AJAX dan cara memanfaatkannya dalam pengembangan web.

AJAX pertama kali di perkenalkan oleh Jesse James Garrett pada tulisannya yang berjudul [AJAX: A New Approach To Web Applications](#).

AJAX bukan merupakan bahasa pemrograman baru, namun hanya sebuah teknik pemanfaatan Javascript dalam mengontrol class object XMLHttpRequest untuk me-Refresh / mengupdate content dalam halaman web tanpa melakukan Reload keseluruhan halaman web seperti pada Metode Tradisional, AJAX sendiri merupakan singkatan dari “Asynchronous JavaScript And XML”.

Kelebihan utama AJAX sendiri terletak pada pemanfaatan class object XMLHttpRequest untuk berkomunikasi dengan Web Server dalam melakukan request content, berikut penggunaan Javascript dalam membuat class object XMLHttpRequest :

Untuk membuat class object pada browser Firefox, Safari, IE7, Mozilla, dsb.

```
1. oAJAX = new XMLHttpRequest();
```

Untuk membuat class object pada browser IE versi terbaru.

```
2. oAJAX= new ActiveXObject('MSXML2.XMLHTTP');
```

Untuk membuat class object pada browser IE versi lama.

```
3. oAJAX = new ActiveXObject('Microsoft.XMLHTTP');
```

Untuk dapat berjalan pada semua browser (Cross Browser), kita dapat mengabungkan kesemua script tersebut kedalam sebuah function pembuat class object XMLHttpRequest seperti berikut :

```
1. function createRequest() {
2. var oAJAX = false;
3. /*@cc_on @*/
4. /*@if (@_jscript_version >= 5)
5. try {
6. oAJAX = new ActiveXObject("Msxml2.XMLHTTP");
7. } catch (e) {
8. try {
9. oAJAX = new ActiveXObject("Microsoft.XMLHTTP");
10. } catch (e2) {
11. oAJAX = false;
12. }
13. }
14. @end @*/
15.
16. if (!oAJAX && typeof XMLHttpRequest != 'undefined') {
```

```

17.     oAJAX = new XMLHttpRequest();
18. }
19. if (!oAJAX) {
20.     alert("Error saat membuat XMLHttpRequest!");
21. }else{
22.     alert("XMLHttpRequest sukses dibuat!");
23. }
24. return oAJAX;
25. }
```

Mula-mula function diatas akan mencoba membuat class object XMLHttpRequest dengan sintaks IE versi terbaru, jika gagal maka akan di coba menggunakan sintaks pd IE versi sebelumnya jika masih gagal maka akan di coba menggunakan sintaks pembuatan pada browser Mozilla, Firefox dsb.

Dengan pengabungan diatas di pastikan class object XMLHttpRequest akan terbentuk dan function akan mengembalikan class object yang telah berhasil dibuat (kecuali ada ada browser baru 😊).

Untuk view example demo [click disini](#)

Setelah object XMLHttpRequest terbentuk selanjutnya kita dapat manfaatkan object tersebut untuk berkomunikasi dengan server, berikut contoh pemanfaatan class object XMLHttpRequest untuk meminta content data dari server.

Untuk merequest data dari server pada contoh berikut di gunakan function *requestContent* yang akan di panggil saat sebuah link di click, detail script function yang digunakan seperti berikut :

```

26.     function requestContent() {
27.         oRequest = createRequest();
28.         var url = "dataLoad.html";
29.
30.         // Buka komunikasi dengan server
31.         oRequest.open("GET", url, true);
32.
33.         // menunggu respon dari server
34.         // jika sudah didapat respon dari server, maka hasil respon akan
            di tampilkan pada DIV dengan ID="divContent"
35.         oRequest.onreadystatechange = function () {
36.             document.getElementById("divContent").innerHTML =
37.                 <p align="center">Menunggu Respon server
38.
39.             ";
40.
41.             if (oRequest.readyState == 4) {
42.                 // baca data respon dari server
43.                 var response = oRequest.responseText;
44.                 document.getElementById("divContent").innerHTML = response;
45.             }
46.         }
47.
48.         oRequest.send(null);
49.     }</p>
```

Function diatas pertama kali akan membuat class object XMLHttpRequest dengan memanfaatkan function *createRequest()* yang ada sebelumnya, hasil pembuatan object akan ditampung kedalam variabel object oRequest.

setelah object XMLHttpRequest terbentuk, berikut akan dilakukan komunikasi dengan request dengan server untuk meminta data dari file dataLoad.html dengan GET method.

Sintaks open request sebagai berikut :

1. oXML.open(<method>, <url\_file>, <async\_status>);</async\_status></url\_file></method>
  - method : POST atau GET
  - url\_file : lokasi URL file yang akan di request
  - async\_status : TRUE atau FALSE

#### **catatan Asynchronous status:**

- Status TRUE :  
Javascript akan melanjutkan proses secara normal sementara menunggu respon dari server diterima
- Status FALSE:  
akan menunggu hingga respon dari server, selama respon belum di dapat proses yang ada berikutnya akan di hold hingga respon dari server di terima.

Setelah melakukan request kepada server function akan menunggu respon dari server

1. oRequest.onreadystatechange

selama menggu respon dari server, bagian script berikut pd function akan dijalankan

1. document.getElementById("divContent").innerHTML="
2. <p align="center">Menunggu Respon server
- 3.
4. ";
5. if (oRequest.readyState == 4) {
6. // baca data respon dari server
7. var response = oRequest.responseText;
8. document.getElementById("divContent").innerHTML = response;
9. }
10. }</p>

#### **ReadyState status:**

0: uninitialized → Open status belum dijalankan

1: loading → Status request belum di jalankan

2: loaded → Status request sudah di kirimkan, dan menunggu respon dari server.

3: interactive → Respon dari server sedang dalam proses download.

4: completed → Respon data dari server telah selesai di download.

Function akan menunggu hingga status respon dari server `oRequest.readyState == 4`, nilai 4 tersebut berarti respon telah selesai diberikan oleh server, setelah respon selesai di kirimkan hasil respon di tampung kedalam variabel bernama `response`, proses menampung data respon ini menggunakan sintaks berikut :

```
1. var response = oRequest.responseText;
```

Dan selanjutnya data hasil respon di tampilkan pada div dengan nama “`divcontent`” yang telah di persiapkan sebelumnya, untuk menampilkan hasil respon ini menggunakan bantuan DOM (document object model) berikut :

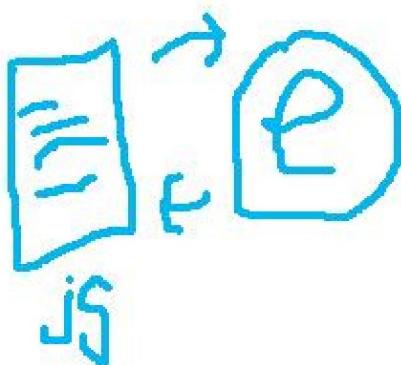
```
1. document.getElementById("divContent").innerHTML = response;
```

Untuk view example demo [click disini](#)

Sampai disini baru terlihat pemanfaatan Javascript dalam menggunakan object XMLHttpRequest saja, data yang di berikan oleh server masih berupa file content statis dan belum memanfaatkan XML data respon.

Semua file contoh ini dapat di [download disini](#)

## 12. Tutorial AJAX, teknik membuat aplikasi web lebih interactive



Sudah lumayan lama saya menggunakan aplikasi web yang menggunakan AJAX, tetapi baru sekarang saya peduli dan ingin tahu lebih dalam tentang AJAX. Jadi saya coba coba browsing sana sini untuk mendapat informasi lebih lanjut mengenai AJAX. Ini adalah rangkuman yang saya dapat dari perjalanan saya untuk mengerti AJAX, saya tulis dalam bahasa Indonesia saja yah. Apa sih AJAX itu ?

AJAX adalah suatu teknik untuk membuat aplikasi web kita menjadi interactive dan attractive, jadi AJAX bukanlah suatu bahasa pemrograman. Btw AJAX itu kependekan dari Asynchronous JavaScript and XML.

Lalu bagaimana bisa lebih interactive dan attractive ?

Dengan AJAX, aplikasi web kita bisa langsung berkomunikasi dengan server untuk meminta sebuah data atau informasi tanpa aksi dari user. Jadi tanpa reload halaman atau klik tombol submit atau apapun, user dari aplikasi web bisa kita beri beragam informasi atau hal hal lain untuk memperindah aplikasi web kita.

Perlu di ingat bahwa ajax hidup berdasarkan web standard yang sudah ada seperti HTML, JavaScript dan web standard lainnya yang saya gak tau. Jadi ajax ini kemungkinan sudah didukung oleh banyak browser ternama di dunia internet, tapi masih ada kemungkinan ajax tidak jalan dibeberapa browser lawas atau yang bandel tidak mengikuti standard.

Btw detail spec saya:

1. Browser Firefox 2.0
2. Os Windows Vista

Rencananya kita akan membuat aplikasi kalkulator kuadrat, sesaat user memasukan sebuah angka maka sesaat itu juga aplikasi langsung mengembalikan nilai kuadratnya.

Pertama yang kita harus lakukan adalah membuat user interface aplikasi kita dulu. Rencananya di aplikasi kita ini akan ada sebuah inputan untuk meminta data dari user, yang nantinya

langsung memberikan hasil secara tiba tiba.

```
1 <html>
2 <body>
3
4 <form name="formku">
5 <input type="text" name="angka" />
6 <span id="texthasil"></span>
7 </form>
8
9 </body>
▶ 10 </html>
```

Penjelasan :

<input type="text" name="angka"> adalah kode untuk membuat inputan  
<span id="texthasil"></span> adalah tempat untuk mencetak hasil balasan dari server

Kedua kita akan buat file php yang akan mengembalikan nilai kuadrat dari inputan user, kebetulan di computer saya hanya ada server php jadi saya menggunakan yang seadanya di computer saya. Untuk anda yang terbiasa atau mempunyai banyak pemrograman web lain selain php dipersilahkan mencoba sendiri.

```
1 <?php
2
3 $angka=$_GET['angka'];
4 print $angka*$angka;
5
▶ 6 ?>
```

Catatan : nama file php diatas saya namakan **serv.php**

Ketiga kita akan membuat fungsi javascript yang menggunakan object **HttpRequest** disitulah ajax bermain. Versi lengkapnya

```

1 <html>
2 <body>
3
4 <script type="text/javascript">
5
6
7 function ajaxFunction(angka)
8 {
9
10    var xmlhttp=new XMLHttpRequest();
11
12    var url="serv.php?angka="+angka;
13
14    xmlhttp.onreadystatechange=function()
15    {
16        if (xmlhttp.readyState==4)
17        {
18            document.getElementById("texthasil").innerHTML=xmlhttp.responseText;
19        }
20    }
21
22    xmlhttp.open("GET",url,true);
23    xmlhttp.send(null);
24
25 }
26
27 </script>
28
29 <form name="formku">
30
31 <input type="text" name="angka" onkeyup="ajaxFunction(this.value)" />
32 <span id="texthasil"></span>
33
34 </form>
35
36 </body>
37 </html>

```

(klik untuk melihat lebih lengkap)

Penjelasan :

**Var xmlhttp=new XMLHttpRequest();**

membuat object HttpRequest yang akan digunakan untuk mengirim data ke server, untuk membuat object tersebut bisa berbeda beda tergantung dari browsernya. Ada bagusnya kalian melakukan check dan ricek dahulu browser apa yang digunakan, atau gunakan script dibawah untuk menemukan object yang tepat untuk browser pengguna.

```

var xmlhttp=null;
try
{
// Firefox, Opera 8.0+, Safari
xmlhttp=new XMLHttpRequest();
}
catch (e)
{
// Internet Explorer
try
{
xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
}

```

Catatan : saya mengkopi script diatas dari aplikasi saya yang sudah berjalan, seinget saya script itu juga saya copy dari suatu tempat, tapi sayangnya saya lupa. X)

**var url="serv.php?angka="+angka;**

Aplikasi php saya yang meminta inputan sau parameter, yaitu angka. Nantinya akan mengembalikan nilai kuadrat dari parameter yang dimasukan.

**xmlHttp.onreadystatechange=function()**

Disini akan menjalankan kode inline fungsi di bawahnya, yang akan merekam kondisi apa saja yang terjadi di saat aplikasi dijalankan, contoh aplikasi hanya merekam kondisi status readyState sama dengan 4.

**xmlHttp.readyState==4**

ini menyatakan aplikasi sukses mengembalikan nilai, dibawah adalah beberapa kondisi yang sering saya gunakan untuk aplikasi saya.

#### **Status Keterangan**

- 2 Request sudah dikirim
- 3 Request dalam proses
- 4 Request komplit

**document.getElementById("texthasil").innerHTML=xmlHttp.responseText;**

Kode ini menampilkan response dari server kedalam <span id="texthasil"></span>

## 13. Membuat File pdf dengan PHP

PHP tidak hanya berguna untuk menampilkan halaman web berbentuk html format. PHP juga bisa digunakan untuk menulis halaman dalam format PDF. Article ini akan membahas bagaimana membuat dokumen PDF dengan menggunakan PDF Class

Untuk membuat halaman PDF dengan php script dan FPDF Class, lakukan langkah berikut :

1. Download FPDF class dari <http://www.fpdf.org/>
2. Extract, dan simpan di folder web anda
3. Buat file php. Misalnya nama file tersebut pdf.php dengan isi sebagai berikut :

```
require('fpdf/fpdf.php');
$pdf=new FPDF();
$pdf->AddPage();
$pdf->SetFont('Arial', 'B', 16);
$pdf->Cell(40, 10, 'Hello World!');
$pdf->Output();
?>
```

4. Panggil script (pdf.php) dengan browser Anda.
5. Anda bisa juga menggunakan fpdf metode lainnya seperti Line, Rectangle, dan yang lainnya.

## 14. Membuat URL dengan Alamat PHP Menjadi HTML

Dulu banget waktu masih newbie belajar tentang HTML, saya banyak mengunjungi situs-situs tutorial seperti about.com. Contoh URL tutorialnya itu seperti ini nih bentuknya:

[http://php.about.com/od/learnphp/p/learn\\_php\\_free.htm](http://php.about.com/od/learnphp/p/learn_php_free.htm)  
[http://php.about.com/od/phpbasics/p/database\\_links.htm](http://php.about.com/od/phpbasics/p/database_links.htm)

Melihat bentuk url yang seperti itu, saya pikir isi server about.com rumit banget. Tutorialnya kan ada banyak. Berarti mereka rajin sekali ya.. membuat halaman html sebanyak itu dalam folder-folder sebanyak itu. Belum lagi kalau desainnya berubah. Wah. Kalau ada 1000 halaman tutorial, berarti mereka mesti meng-update 1000 file dalam sehari sekaligus. Haha :D. Tapi ya saya maklumi saja. Namanya juga situs gede. Pasti sudah punya modal hostingan gede pula untuk menampung ribuan halaman HTML (berpikir polos mode on ^^).

Tapi sejak 3 tahun yang lalu (huyuh.. telat banget) saya baru nyadar kalo 'halaman-halaman HTML' itu aslinya halaman dinamis (ASP atau PHP) yang di mod rewrite. ^^

Bentuk mudahnya mungkin seperti ini:

<http://www.domain.com/article.php?title=hello-world>

menjadi

<http://www.domain.com/hello-world.html>

atau dari

<http://www.domain.com/article.php?title=hello-world&category=uncategorized>

menjadi

<http://www.domain.com/uncategorized/hello-world.html>

Intinya, dengan mod rewrite, kita bebas menentukan format bagaimana bentuk url itu jadinya.

Kebanyakan blog dan situs-situs besar biasanya memasang judul artikel (atau judul produk jika jenis websitenya e-commerce) sebagai URLnya.

Ada banyak tutorial di internet yang membahas tentang mod rewrite. Disini saya coba uraikan cara membuat mod rewrite yang menurut saya paling mudah dan paling sering saya pakai.

Sebagai study case, mari kita membuat mod rewrite untuk halaman detail sebuah portal berita yang (misalnya) anda punya.

First of all, anggaplah kita sudah mempunyai database yang menampung berita-berita yang akan ditampilkan. Asumsikan bahwa kita akan menampilkan berita-berita tersebut melalui sebuah halaman PHP yang bernama **berita.php**.

Jika biasanya berita ditampilkan dalam bentuk URL berikut:

<http://www.situsberita.com/berita.php?judul=jakarta-heboh-kebanjiran>

## **lagi&kategori=bencana-alam**

maka kali ini kita ingin supaya format berita tampil dengan url berikut:

**<http://www.situsberita.com/bencana-alam/jakarta-heboh-banjir-lagi.html>**

Langkah-langkahnya yaitu:

1. mula-mula, buka hostingan situs berita tsb dengan ftp atau file manager.
2. download file .htaccess dari folder public\_html (atau www)

Jika tidak ketemu filenya, maka bisa bikin sendiri di notepad (kosongan), lalu simpan dengan nama **.htaccess** (tanpa tambahan nama lain atau extension lain) dan **save as typenya all files**.

3. Edit file tersebut dengan text editor favorit. Saya biasanya pakai notepad++ :)
4. Tambahkan script berikut kedalamnya.

```
RewriteEngine on
```

```
RewriteBase /home/user/public_html/
RewriteRule ^(.*)/(.*)\.html /berita.php?judul=$1&kategori=$2
```

*note: modify path pada baris ke 3. Pada baris ke 4, format url yang diinginkan berada di sebelah kiri (diawali dengan ^), sedangkan format url yang sebenarnya berada di sebelah kanan (diawali dengan /). Kiri dan kanan dipisahkan dengan satu spasi. \$1 untuk memanggil nilai wildcard(.\* ) yang pertama, sedangkan \$2 untuk memanggil (.\* ) yang kedua, dst. Jika ingin menambahkan rule baru, tambahkan saja langsung di bawahnya lagi tanpa harus mengulang baris 1-3.*

5. Save
6. upload lagi (overwrite)

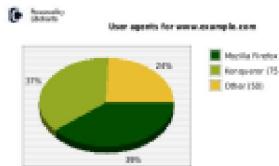
Sudah selesai :). Silakan buka internet browser anda, dan ketik langsung URL dengan format berikut di address bar:

**<http://www.situsberita.com/nama-kategori/judul-berita.html>**

Nah. Mudah-mudahan sukses. Selebihnya tinggal tugas anda untuk merubah nilai href didalam tag <a> di seluruh bagian situs tersebut supaya urlnya menggunakan alamat baru. Jangan lupa di **urlencode()** dulu. Selanjutnya dipanggil dengan **urldecode()**.

Penggunaan mod rewrite membantu menaikkan posisi situs anda pada search engine. Karena itu, banyak pakar SEO yang mengharuskan penggunaan mod rewrite ini pada setiap situs.

## 15. Membuat chart dengan Codeigniter dan php libchart



Ketika kita membuat sebuah laporan maka satu hal yang harus di buat adalah char. Sudah library PHP yang ada dan dapat digunakan untuk membuat chart tersebut. PHP libchart adalah salah satu library untuk membuat sebuah chart pada PHP berbasis GD. Untuk menggabungkannya saya buat sebuah pluggin chart sebagai factory dari lib chart tersebut. adapun langkah-langkah yang dilakukan adalah

### Langkah 1 : Download PHP libchart

Download library di <http://naku.dohcrew.com/libchart/pages/introduction/>

### Langkah 2 : Extract PHP libchart

extract libchart pada direktori system/application/pluggin/

### Langkah 3 : Buat file chart\_pi.php

buat file chart\_pi.php yang berisi

```
1  <?
2  if (!defined('BASEPATH'))
3  exit('No direct script access allowed');
4  require_once("libchart/classes/libchart.php");
5
6 /**
7 * @author Ibnu Daqiqil Id
8 * @param string      $title   Judul Chart
9 * @param array       $data    Data array dengan dua dimensi (key,value)
10 * @param integer     $x       Lebar chart
11 * @param integer     $y       Tinggi Chart
12 * @param string      $type    tipe output chart
13 (bar_vertikal,bar_horizontal,pie)
14 * @param boolean     $render  Apakah di render ke file?
15 */
16 function
17 create_bar_chart($title,$data,$x=500,$y=300,$type="bar_vertikal",$render_f
18 ile=False)
19 {
20     if ("bar_horizontal"==$type)
21         $chart = new HorizontalBarChart($x,$y);
22     else if ("bar_vertikal"==$type)
23         $chart = new VerticalBarChart($x,$y);
24     else
25         $chart = new PieChart($x,$y);
26
27     $dataSet = new XYDataSet();
28     foreach ($data as $value)
29     {
30         $dataSet->addPoint(new Point($value['key'],
```

```

31 $value['value']));
32     }
33     $chart->setDataSet($dataSet);
34     $chart->setTitle($title);
35     if (! $render_file)
36         return $chart->render();
37     else
38         return $chart->render($render_file);
39     }
40 ?>

```

Cara penggunaannya adalah

```

class chart_demo extends Controller {
    function chart_demo()
1        parent::Controller();
2    }
3    function index()
4    {
5        echo "<img
6 src='index.php/chart_demo/generate_chart_manual'><br />" ;
7        echo "<img
8 src='index.php/chart_demo/generate_chart_database'>" ;
9
10    }
11
12    function generate_chart_manual()
13    {
14        $data[0]["key"]="key 1";
15        $data[0]["value"]="17";
16        $data[1]["key"]="key 2";
17        $data[1]["value"]="26";
18        $data[2]["key"]="key 4";
19        $data[2]["value"]="37";
20        $data[3]["key"]="key 4";
21        $data[3]["value"]="7";
22
23        echo create_bar_chart("Alasan Pekerjaan", $data, 450, 250);
24    }
25    function generate_chart_database()
26    {
27        $this->load->plugin('chart');
28        $this->load->database();
29        $this->db->select (" job_description as key,
30 count(job_description) as value");
31        $this->db->group_by('job_description');
32        $query=$this->db->get('jobreasongraph');
33        echo create_bar_chart("Alasan Pekerjaan", $query-
34 >result_array(), 450, 250);
35    }
}

```

## 16. Script PHP untuk Membuat Dokumen RTF

Artikel ini terinspirasi oleh requestnya mas “Kereayem” yang menanyakan bagaimana cara membuat dokumen DOC dengan script PHP di salah satu komentarnya dalam blog ini. Namun maaf mas... karena sepertinya dokumen DOC sulit dimanipulasi dengan PHP, sehingga sebagai gantinya adalah dokumen dalam bentuk RTF. Setelah cari referensi sana sini terkait dengan hal ini, akhirnya ketemu juga ilmunya. Nah... sekarang saatnya saya berbagi ilmu tersebut melalui blog ini.

OK... dalam artikel ini akan dijelaskan bagaimana cara mengenerate dokumen RTF (Rich Text Format) dengan script PHP. Mengapa RTF? kenapa bukan DOC nya MS. Word? Yup... good question. Konten dokumen RTF lebih mudah untuk dimanipulasi dibandingkan dokumen DOC. Hal ini dikarenakan dalam dokumen DOC terdapat unsur grafis, dan bukanlah text based. Untuk menunjukkan hal ini, cobalah dokumen DOC ini Anda buka melalui text editor seperti NOTEPAD, atau Crimson Editor dll, pastilah isinya semrawut dan di sana sini tampil kode-kode yang tidak bisa dibaca. Namun beda halnya bila Anda membuka file RTF dengan text editor, pastilah isinya tidak semrawut dan dapat dibaca. Ya... dokumen RTF merupakan dokumen berbasis text based. Oleh karena itu, isi dokumen RTF ini dapat kita manipulasi dengan script untuk mendapatkan dokumen output yang diinginkan.

Dalam contoh ini, studi kasus yang diambil adalah membuat surat undangan dimana nama orang yang diundang, alamatnya, serta tempat dan waktu undangan akan diinputkan melalui suatu form. Setelah form ini diproses, selanjutnya akan muncul surat undangan berbentuk dokumen RTF dengan isi sesuai dengan data yang dimasukkan via form tadi.

Untuk membuat studi kasus di atas, apa yang dilakukan pertama kali? Hal pertama yang Anda lakukan adalah membuat template dokumen untuk surat undangannya. Caranya adalah bukalah MS. Word, lalu ketikkan sebarang format surat undangan, misalnya seperti di bawah ini:

Hal : Undangan

Kepada Ykh.  
Bp/Ibu %NAMA%  
Di %%ALAMAT%%

Dengan hormat,  
Mengharap kehadiran Bapak/Ibu untuk menghadiri rapat koordinasi  
yang akan diadakan pada:

Tanggal : %%TGL%%  
Tempat : %%TEMPAT%%  
Waktu : %%WAKTU%%

Demikian undangan ini dibuat, atas kehadirannya disampaikan  
terimakasih.

Surakarta, 5 Nopember 2008

Sekretaris

Perhatikan template di atas! Dalam template terdapat penanda untuk masing-masing item, misalnya untuk item nama yang diundang diberi tanda %%NAMA%%, tanggal undangannya diberi tanda %%TGL%% dsb. Tanda-tanda inilah yang nantinya akan digantikan dengan data yang disubmit via form. Apakah tandanya harus diapit dengan %% ? Sebaiknya ya.. karena tanda % dalam RTF tetap ditulis apa adanya. Maksudnya, bila Anda menulis % di dokumen RTF, lalu membukanya di text editor maka tanda % tadi tidak akan berubah. Namun beda hasilnya bila Anda menggunakan tanda #, misalnya. Bila Anda ketika # di dokumen, lalu membukanya di text editor, maka tanda ini akan dinyatakan dalam bentuk yang lain.

Setelah Anda buat dokumen templatenya, sekarang Anda simpan template tersebut ke dalam format RTF. Caranya klik menu FILE di MS. Word, pilih SAVE, pada bagian SAVE AS TYPE, pilih (Rich Text Format) RTF. Misalkan dokumen tersebut kita simpan dengan nama file surat.rtf.

Nah... untuk iseng-iseng saja, sekarang coba bukalah surat.rtf ini dengan NOTEPAD, NOTEPAD++, Crimson Editor atau text editor yang lain. Pastilah isi yang tampil adalah sebagai berikut:

(dipotong, karena panjang....)

```
\fs24\lang1033\langfe1033\cgrid\langnp1033\langfnp1033
{\insrsid1652669 Hal\tab : Undangan
\par
\par Kepada Ykh.
\par Bp/Ibu %%NAMA%%
\par Di %%ALAMAT%%
\par
\par Dengan hormat,
\par Mengharap kehadiran Bapak/Ibu untuk menghadiri rapat
koordinasi yang akan diadakan pada:
\par
\par Tanggal\tab : %%TGL%%
\par Tempat\tab : %%TEMPAT%%
\par Waktu\tab \tab : %%WAKTU%%
\par
\par Demikian undangan ini dibuat, atas kehadirannya disampaikan
terimakasih.
\par
\par Surakarta, 5 Nopember 2008
\par
\par Sekretaris
\par
\par
\par Rosihan Ari Yuana
\par }}
```

Catatan: tempate di atas dibuat dengan MS. Word 2003. Bisa jadi isi dokumen RTF di atas berbeda untuk setiap versi MS. Word.

Anda perhatikan isi template, bahwa tanda-tanda untuk setiap item yang kita buat tadi masih terlihat. Sehingga tugas kita selanjutnya adalah melakukan replacing string untuk masing-masing penanda dengan data yang diisikan via form. Itu adalah ide dasarnya.

OK... template surat sudah dibuat, langkah selanjutnya adalah membuat form untuk mengisi data surat. Ini dia contoh formnya.

```
<h1>Buat Surat Undangan</h1>
1 <form method="post" action="surat.php">
2 <table>
3 <tr><td>Nama Penerima Surat</td><td><input type="text"
4 name="nama"></td></tr>
5 <tr><td>Alamat Penerima Surat</td><td><input type="text"
6 name="alamat"></td></tr>
7 <tr><td>Tanggal Acara</td><td><input type="text" name="tanggal"></td></tr>
8 <tr><td>Tempat Acara</td><td><input type="text" name="tempat"></td></tr>
9 <tr><td>Waktu Acara</td><td><input type="text" name="waktu"></td></tr>
10 <tr><td></td><td><input type="submit" name="submit"
11 value="Submit"></td></tr>
12 </table>
</form>
```

Berikutnya, kita buat script surat.php untuk memproses form. Di dalam script PHP ini pastilah terdapat perintah untuk melakukan replacing string masing-masing penanda dengan data string dari form. Apa perintah yang bisa digunakan? Untuk replacing ini kita bisa gunakan str\_replace().

surat.php

```
1 <?php
2
3 // membaca data dari form
4
5 $nama = $_POST['nama'];
6 $alamat = $_POST['alamat'];
7 $stanggal = $_POST['tanggal'];
8 $waktu = $_POST['waktu'];
9 $tempat = $_POST['tempat'];
10
11 // membaca isi dokumen tempate surat.rtf
12 // isi dokumen dinyatakan dalam bentuk string
13
14 $document = file_get_contents("surat.rtf");
15
16 // mereplace tanda %%NAMA% dengan data nama dari form
17 $document = str_replace("%%NAMA%%", $nama, $document);
18
19 // mereplace tanda %%ALAMAT% dengan data alamat dari form, dst
20 $document = str_replace("%%ALAMAT%%", $alamat, $document);
21
22 $document = str_replace("%%TGL%%", $stanggal, $document);
23 $document = str_replace("%%TEMPAT%%", $tempat, $document);
```

```
24 $document = str_replace("%%WAKTU%%", $waktu, $document);  
25  
26 // header untuk membuka file output RTF dengan MS. Word  
27 // nama file output adalah undangan.rtf  
28  
29 header("Content-type: application/msword");  
30 header("Content-disposition: inline; filename=undangan.rtf");  
31 header("Content-length: " . strlen($document));  
32 echo $document;  
33  
34 ?>
```

Begitulah cara untuk mengenerate dokumen RTF dengan script PHP. Memang dalam contoh ini data yang disisipkan ke dokumen RTF bersifat single data. Untuk multiple data, seperti misalnya menampilkan data-data hasil query ke dalam bentuk tabel pada dokumen RTF, hal ini agak sulit dilakukan. Hal ini dikarenakan format tabel dalam dokumen RTF sangat kompleks bentuknya. Tapi bukan berarti tidak mungkin dibuat. Bila ada di antara Anda yang sudah berhasil membuat dokumen RTF dengan multiple data ini, dapat dishare ilmunya ke saya 😊

OK... selamat mencoba

NB:

Anda dapat mendownload script di atas (termasuk dokumen RTF template suratnya), di bawah ini

[\*\*Download Script\*\*](#)

## 17. Tips Membuat Web Full Ajax Sederhana

---

Fenomena Ajax yang menjanjikan pengembangan aplikasi web menjadi lebih dinamis dan interaktif membuat para developer pun terus menerus melakukan eksperimen. JavaScript yang menjadi unsur dasar Ajax pun terus menerus dikembangkan dan hampir seluruh website sudah pasti menggunakan JavaScript untuk membuat setiap halamannya lebih hidup dan menarik.

Yang membuat Ajax menarik adalah kita dapat mengupdate satu atau lebih dari element di halaman web kita tanpa perlu reload halaman. Bila reload dapat dihindarkan tentunya kita berhemat waktu pengambilan data.

Seiring berkembangnya Ajax, banyak pertanyaan dan usulan muncul kepada saya tentang bagaimana kalau kita buat saja 1 web yang seluruh fasilitasnya merupakan Ajax, jadi tidak ada lagi berpindah-pindah halaman karena semua proses dilakukan pada 1 file. Bagaimana mungkin? namun di dalam prinsip saya “*kalau tidak dicoba maka belum boleh bilang tidak mungkin*” bukan?? ^\_^

Disini saya punya tips dasar membuat website berpindah-pindah halaman/proses dengan Ajax dan kita akan lakukan semua proses di 1 file saja yaitu index.html.

Pertama-tama siapkan 2 buah halaman (atau lebih) untuk menjadi **sample content** yang akan bergantian di load sesuai request dari index.html.

```
<h2>Hello this is content 1</h2>
1 Loren in sum sit amet this is just a sample text for web development loren
2 in sum sit amet this is just a sample text for web development loren in sum
   sit amet this is just a sample text for web development.
1 <h2>This is content 2</h2>
2 <?
3         foreach($_SERVER as $key=>$value){
4             echo $key . ' : ' . $value . '<br />';
5         }
6 ?>
```

Lalu buat file **index.html** dengan isi sederhana seperti ini:

```
39 <table width="80%" border="1">
40   <tr valign="top">
41     <td width="15%">
42       <a href="javascript:loadModule('content1') ">Content 1</a><br />
43       <a href="javascript:loadModule('content2') ">Content 2</a>
44     </td>
45     <td width="85%" id="tdContent"></td>
46   </tr>
47 </table>
```

Sebelum melanjutkan topik, saya akan jelaskan apa pengertian module disini dan bagaimana 1 file dapat memproduksi berbagai module, jawabannya adalah kita bermain di lokasi dokumen (`document.location`). Coba perhatikan URL berikut:

<http://yourweb/index.html#content1>  
<http://yourweb/index.html#content2>

Pada url diatas string '`content1`' dan '`content2`' adalah **nama module** dan apabila url ini dipanggil akan ada JavaScript yang mendeteksi nama module dari lokasi dokumen (URL) dan akan mengeksekusi perintah-perintah yang harus dikerjakan apabila module ini dipanggil. Cukup jelas??

Untuk memanipulasi lokasi dokumen (URL) dan mendapatkan nama module tersebut saya sudah menyediakan class sederhana dan simpan nama class tersebut dengan nama **document.class.js**:

```
var T10Doc = Class.create({
  _splitLocation: function() {
    var docLoc = new String(document.location);
    return docLoc.split('#'); // pecah lokasi document berdasarkan karakter '#'
  },
  getModule: function() {
    var arr = this._splitLocation();
    if(arr.length == 2)
      return arr[1]; // nama module adalah string setelah karakter '#'
    else return null;
  },
  saveModule: function(moduleName) {
    var arr = this._splitLocation();
    /*
      Apabila sebelumnya sudah ada nama module maka ganti nama
      module tersebut dengan nama module baru namun apabila lokasi
      tidak terdapat nama module maka tambahkan #namamodule
      setelah string lokasi dokumen
    */
    if(arr.length == 2) document.location = arr[0] + '#' + moduleName;
    else document.location += '#' + moduleName;
  }
});
```

Langkah terakhir adalah sisipkan script di bawah ini diantara tag head index.html yang telah anda buat sebelumnya:

```
4 <script language="javascript" src="prototype.js"></script>
5 <script language="javascript" src="document.class.js"></script>
6 <script language="javascript">
7   var doc = new T10Doc();
8   function loadModule(moduleName) {
9     switch(moduleName) {
```

```

10         case 'content1' :
11             // load content 1 dan tampilkan isinya pada element
12 yang disediakan
13             new Ajax.Updater('tdContent', 'content1.htm', {method:
14 'GET'});
15             break;
16         case 'content2' :
17             // load content 2 dan tampilkan isinya pada element
18             new Ajax.Updater('tdContent', 'content2.php');
19             break;
20         default:
21             // apabila nama module tidak terdefinisi, kosongkan
22             $('tdContent').update(' ');
23         }
24         doc.saveModule(moduleName); // ubah url menjadi [lokasi
25 dokumen]#[nama module]
26     }
27 window.onload = function() {
28     /*
29         Apabila browser di refresh dan terdapat nama module pada
30 lokasi dokumen
mendapatkan
31         maka eksekusi module tersebut. Fungsi doc.getModule() untuk
32         nama module saat ini.
33     */
34     loadModule(doc.getModule())
35 }
</script>

```

Hasilnya adalah ketika anda menekan salah satu link content maka ada 2 hal yang terjadi yaitu isi dari kolom sebelah kanan berubah sesuai content yang ditentukan dan URL (lokasi dokumen) anda pun berubah dan menyimpan nama module dengan format [lokasi dokumen]#[nama module]. Apabila anda refresh halaman anda, maka url tidak akan berubah dan dengan sendirinya module terakhir sebelum anda me-refresh browser akan tereksekusi kembali.

Untuk melihat demo dari artikel ini anda dapat melihat di [link ini](#).

Download contoh aplikai lengkap dari artikel ini di [link ini](#).

Semoga artikel saya ini dapat membantu anda dan menambah wawasan anda... Terima kasih...

^ ^  
—

## 18. Membuat PHP

Di bagian ini kita akan membahas

- Define
- PHP Decision Making

Ok kita mulai

### - Define

di sini saya akan jelaskan sedikit tentang define:  
define mirip dengan, "string":

```
-----  
<?php  
define("HELLO", "Hello world!");  
echo HELLO;  
?>
```

-----  
Nanti hasilnya: "Hello world."

### - PHP Decision Making

PHP Decision Making yang akan saya jelaskan kali ini adalah yang sering di pakai dalam programing website

- if
- else
- elseif

### Penggunaan fungsi "if"

Sebagai penalaran "if" anda samakan dengan "jika"

Contoh:

```
if (kondisi expresi)  
{  
kode anda;  
}
```

Contoh Penerapan:

```
<?php  
if ($user == b_scorpio)  
{  
echo "user anda b_scorpio";  
}  
?>
```

Contoh Penerapan untuk user autentifikasi:

```
<?php  
if ($user == b_scorpio && $password == 8_5c0rp10)  
{  
echo "anda berhasil login";  
}  
?>
```

**Penggunaan fungsi "else"**

Sebagai penalaran anda asumsikan "else" sebagai "jika tidak"

Contoh:

```
if(kondisi expresi)  
{  
kode anda;  
}  
else  
{  
kode anda;  
}
```

Contoh Penerapan:

```
<?php  
if ($user == b_scorpio)  
{  
echo "user anda b_scorpio";  
}  
else  
{  
echo "anda bukan b_scorpio";  
}  
?>
```

Contoh Penerapan untuk user autentifikasi:

```
<?php  
if ($user == b_scorpio && $password == 8_5c0rp10)  
{  
echo "anda berhasil login";  
}  
else  
{  
echo "anda tidak bisa login";  
}  
?>
```

**Penggunaan fungsi "elseif"**

"elseif" di gunakan untuk statement "if" lebih dari satu

### Contoh Penerapan untuk user autentifikasi:

```
-----  
<?php  
if ($user == b_scorpio)  
{  
echo "anda login sebagai $user";  
}  
  
elseif($user == alk)  
{  
echo "anda login sebagai $user";  
}  
  
elseif($user == johan)  
{  
echo "anda login sebagai $user";  
}  
  
else  
{  
echo "anda tidak bisa login";  
}  
?>
```

Implementasi dari Penerapan untuk user autentifikasi, lebih lanjut Anda bisa lihat di bagian PHP Tutorial

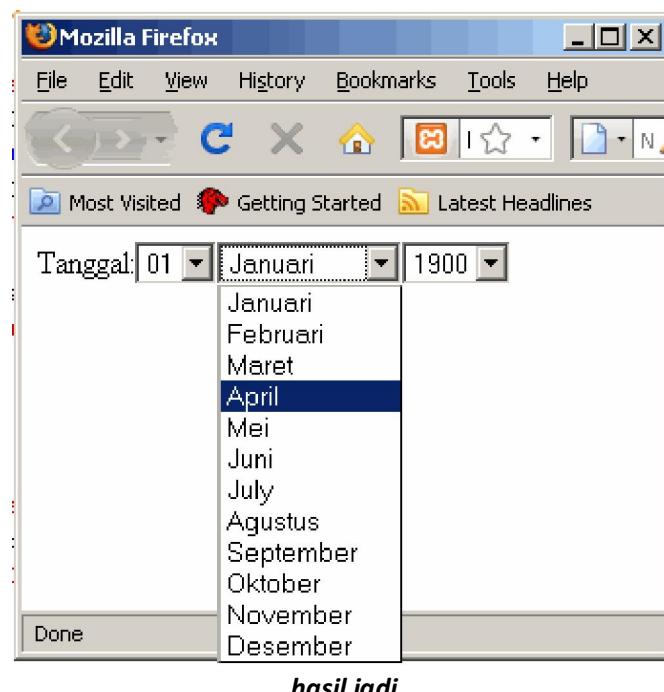
## 19. Membuat pilihan Tanggal dengan ComboBox pada PHP

Biasanya ketika kita membuat suatu web, pastinya terdapat unsur tanggal bukan? Nah, disini saya akan menuliskan bagaimana cara membuatnya dengan bahasa PHP. Salah satu cara untuk memilih tanggal adalah menggunakan ComboBox. Tujuan dengan menggunakan ComboBox adalah untuk mempermudah user dalam memilih pilihan yang sudah disediakan.

Hal ini juga menghindari kesalahan user dalam penulisan suatu format tanggal yang telah disediakan. Untuk lebih jelasnya, lihat pelajari dan coba script berikut ini:

```
<?php  
//array yang digunakan pada ComboBox bulan  
$bln=array(1=>"Januari","Februari","Maret","April","Mei",  
"Juni","July","Agustus","September","Oktober",  
"November","Desember");  
  
//membuat tanggal 1-31 pada ComboBox  
echo "Tanggal:<select name=tanggal>  
<option value=01 selected>01</option>";  
for($tgl=2; $tgl<=31; $tgl++){  
$tgl_leng=strlen($tgl);  
if ($tgl_leng==1)  
$i="0".$tgl;  
else  
$i=$tgl;  
echo "<option value=$i>$i</option>";}  
echo "</select>";  
  
//membuat bulan ComboBox  
echo "<select name=bulan>  
<option value=Januari selected>Januari</option>";  
for($bulan=2; $bulan<=12; $bulan++){  
echo "<option value=$bulan>$bln[$bulan]</option>";}  
echo "</select>";  
  
//Membuat tahun 1900 sampai sekarang pada ComboBox  
$now=date("Y");  
echo "<select name=tahun>  
<option value=1900 selected>1900</option>";  
for($thn=1901; $thn<=$now; $thn++){  
echo "<option value=$thn>$thn</option>";}
```

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



**Download coding:**



## 20. Source Upload dan View Gambar dengan PHP

Disaat kita browsing di internet sering kita jumpai sebuah website dimana kita bisa mengirimkan sebuah file ke server penyedia website. File yang kita kirimkan bisa berbentuk gambar, program, atau file dokumen lainnya. Proses pemindahan file dari komputer kita ke komputer server sering kita namai dengan upload.

Pada kesempatan kali ini, kita akan mencoba membuat sebuah program upload sederhana dengan sebuah inputan data.

1. Buatlah sebuah database dengan nama 'upload\_gambar'. Pada database ini berisi dua field, antara lain : id\_gambar dan gambar. Berikut script pembuatan database 'upload\_gambar':

```
CREATE TABLE `upload_gambar` (
  `id_gambar` int(3) NOT NULL auto_increment,
  `gambar` varchar(25) NOT NULL default '',
  PRIMARY KEY  (`id_gambar`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=31 ;
```

2. Selanjutnya buatlah sebuah direktori dengan nama 'foto' pada direktori dimana anda meletakkan file php yang akan dibuat. Contoh kasus saya menggunakan apache2triad, maka direktori 'foto' saya letakkan di C:\apache2triad\htdocs\upload\gambar dan file php akan saya simpan pada C:\apache2triad\htdocs\upload
3. Setelah database dan direktori 'gambar' kita buat, selanjutnya kita akan membuat antarmuka upload gambar yang akan kita beri nama 'form\_gambar.php'. Script antarmuka upload gambar seperti berikut

```
<form name="input_gambar" method="post" action="simpan_gambar.php" enctype=
"multipart/form-data">
  <table>
    <tr>
      <td width="53">Gambar </td>
      <td width="3">:</td>
      <td width="218"><input name="gambar" type="file" /></td>
    </tr>
    <tr>
      <td colspan="3" align="right"><input name="simpan" type="submit"
      value="Simpan" /></td>
    </tr>
  </table>
</form>
```

4. Langkah selanjutnya kita buat file proses upload, yang akan kita beri nama 'simpan\_gambar.php'. Berikut script 'simpan\_gambar.php'

```

<?
if ($simpan)
{
include_once "koneksi.php";
$save = "INSERT INTO upload_gambar (gambar) VALUES ('$gambar_name')";
$newsimg = "./foto/" . $_FILES[gambar][name];
if ($_FILES[gambar][type] == 'image/gif' || $_FILES[gambar][type] ==
'image/jpeg')
{
    if (copy($_FILES[gambar][tmp_name], $newsimg))
    {
        $hasil=mysql_query($save);
    }
    else
    {
        echo "gagal";
    }
}
else
{
    echo "Bukan type file";
}
if ($hasil)
{
    echo "Upload Berhasil";
    echo "<meta
http-equiv=\"refresh\"content=\"0;url='tampil_gambar.php'\\" />";
}
}
?>

```

5. Untuk menampilkan gambar, kita buat proses tampil gambar. Proses ini akan kita namakan 'proses\_tampil.php'. Script 'tampil\_gambar.php' seperti berikut.

```

<?
include_once "koneksi.php";
$tampil = mysql_query("SELECT * FROM upload_gambar ORDER BY id_gambar");
while ($muncul=mysql_fetch_array($tampil))
{
    echo "$muncul[gambar]<br>";
    echo "<img src=foto/$muncul[gambar] /><br>";
}
?>

```

Selamat Mencoba

## 21. File Upload Progress dengan PHP 5 dan APC



Beberapa hari lalu, secara nggak sengaja gue baca-baca [PHP 5.2.0 Release Announcement](#) (tumben ^^) dan ngelihat kalimat ini di daftar *key features* nya "*Hooks for tracking file upload progress were introduced*". Ho ho ho *was ist das?* Penasaran banget, gue coba tanya-tanya kang google, dan mampir di [satu blog](#) yang bertanya di PHP Development mailing list tentang fungsi tracking yang dijanjikan tersebut. Yang menjawab tidak lain dan tidak bukan adalah [Rasmus Lerdorf](#) sendiri, dan menyatakan bahwa fungsi tersebut masih berada pada level C dan membutuhkan fungsi pendukung untuk digunakan dalam php.

Sudah lama gue coba cari-cari cara untuk bisa menampilkan laporan tentang kemajuan proses sewaktu meng-*upload* file. Gue sebenarnya cari cara, dimana menulis codenya itu murni dalam PHP. Tapi sayangnya kebanyakan yang gue temuin itu [menggunakan cgi dan ditulis dalam Perl](#). Sayangnya gue gak begitu ngerti Perl, dan gue kurang suka sama bahasanya karena "terlalu dinamis", sehingga membuat code yang ditulis cepat untuk menjadi sulit dibaca.

Gue juga sempat coba pakai [Fancyupload](#) yang ditulis oleh [Harald Kirschner](#), sebuah *multi file uploader* yang bekerja menggunakan Flash. Fancyupload ini gue lihat juga dipakai sama [Joomla!](#) CMS. Sayangnya ada 2(dua) kekurangan yang menurut gue cukup penting. Fancyupload ini tidak jalan di mac, dan setiap data yang di upload *content-type* nya pasti "application/octet-stream", bikin sebel. Biasanya gue kalo periksa file yang di upload, gue pasti periksa berdasarkan *extension* dan *content-type* nya, kalo beda biasanya filenya tidak gue terima.

Rasmus sendiri, menurut blog tersebut, langsung menyatakan bahwa dia telah menulis fungsi pendukung tersebut di dalam [Alternative PHP Cache](#), sebuah PHP *library* yang merupakan bagian dari *PHP Extension Community Library (PECL)*. [Contohnya](#), beserta [source code](#) nya pun langsung diberikan. Tapi, menurut gue, sialnya, dia menulis contohnya menggunakan [Yahoo! User Interface Framework](#) dan gak pake komentar. Bukannya gue gak suka sama framework. Tapi menurut gue, kurang bijaksana sekali memberikan contoh code menggunakan framework, gak peduli se-straightforward apapun contohnya. Dan untuk sebuah contoh code, menurut gue lagi, komentar itu luar biasa pentingnya untuk mengerti *step-by-step* apa saja yang terjadi, atau mengapa terjadi, di dalam code tersebut.

Oleh karena itu, gue mau mencoba untuk membuat versi yang lebih sederhana, tanpa framework apapun dan penuh komentar ^^. Dalam kesempatan ini gue mau coba jabarkan dari awal, mulai dari

menginstall APC modul itu sendiri, mengkonfigurasi modul tersebut, hingga menggunakananya.

## Instalasi dan Konfigurasi APC

Untuk menggunakan fungsi *file upload tracker* menggunakan APC, kita butuh php 5.2+. Gue sudah mencoba menginstall APC ini diatas Windows dan Debian *based* Linux. Untuk Windows kita harus mendownload APC library secara [manual](#). [PECL zip package](#) yang disediakan oleh php.net tidak mengikutsertakan library APC dalam distribusinya.

Setelah di download php\_apc.dll tersebut tinggal dimasukkan ke dalam *extension directory* yang telah di definisikan di dalam php.ini. Untuk menginstall APC di dalam Debian based Linux gue menggunakan [tutorial yang ini](#). Untuk install di distribusi linux yang lain dan *Operating System* yang lain silahkan cari di google. Kesuksesan dan kegagalan instalasi biasanya dipengaruhi berbagai macam aspek, yang tidak semuanya gue tahu dan tidak bisa gue jabarkan panjang lebar disini.

Pokoknya setelah kalian berhasil menginstall APC tersebut, maka kita harus mengatur konfigurasinya di dalam php.ini. Yang pertama adalah mendaftarkan *library* tersebut dalam daftar extension supaya dikenal oleh PHP.

1. extension=php\_apc.dll

Di linux tidak berbeda jauh

1. extension=apc.so

Setelah ini, kita harus menyalakan fungsi RFC 1867, dan tergantung kalian ingin mencoba file upload dimana, lokal server atau remote server, ada beberapa perbedaan dalam konfigurasinya supaya nanti codenya bisa di test. Gue akan kasih lihat disini konfigurasi lokal server gue yang gue sadur dari IBM yang [membahas thema yang sama](#).

1. ...
2. post\_max\_size = 400M
3. ...
4. upload\_max\_filesize = 400M
5. ...
6. [APC]
7. apc.rfc1867 = on
8. apc.max\_file\_size = 200M

## 9. ...

Mari kita lihat, gue merubah nilai maximum file size diatas 200M, kenapa? Karena apabila kita mengupload file di lokal server, maka file tersebut ter-upload begitu cepatnya, sehingga APC tidak sempat bereaksi untuk menghitung kemajuannya. Oleh karena itu untuk mengetest di lokal server kita butuh file yang cukup besar, biasanya diatas 150M. Oleh karena itu juga, kita butuh merevisi nilai maximum file size di php.ini. Untuk remote server atau produktif server, kita hanya perlu menyalakan fungsi RFC 1867 saja, dalam arti, kita hanya perlu menambahkan "apc.rfc1867 = on" di dalam php.ini kita. Selebihnya bisa dibiarkan sesuai nilainya standardnya.

Untuk mengetest apakah semua instalasi dan konfigurasi berjalan dengan baik dan benar, maka setelah restart apache kita harung menengok phpinfo(). Apabila disitu terdapat informasi tentang APC seperti *screenshot* dibawah, maka apc sudah terinstall dengan benar dan kita bisa mulai menggunakannya.

apc		
Directive	Local Value	Master Value
apc.cache_by_default	On	On
apc.coredump_unmap	Off	Off
apc.enable_cli	Off	Off
apc.enabled	On	On
apc.file_update_protection	2	2
apc.filters	no value	no value
apc.gc_ttl	3600	3600
apc.include_once_override	Off	Off
apc.localcache	Off	Off
apc.localcache.size	512	512
apc.max_file_size	200M	200M
apc.num_files_hint	1000	1000
apc.report_autofilter	Off	Off
apc.rfc1867	On	On
apc.rfc1867_freq	0	0
apc.rfc1867_name	APC_UPLOAD_PROGRESS	APC_UPLOAD_PROGRESS
apc.rfc1867_prefix	upload_	upload_
apc.shm_segments	1	1
apc.shm_size	30	30
apc.slam_defense	0	0
apc.stat	On	On
apc.stat_ctime	Off	Off
apc.ttl	0	0
apc.user_entries_hint	4096	4096
apc.user_ttl	0	0
apc.write_lock	On	On

Kalau kalian ingin tahu lebih lanjut tentang arti dan fungsi dari masing-masing konfigurasi APC silahkan membaca [dokumentasinya](#). Sekarang kita bisa memulai untuk bereksperimen menampilkan informasi proses upload file. Sekedar informasi, semua code yang akan gue tulis, baik itu php, html atau javascript code, berada dalam satu file, yaitu upload\_progress.php. Hal ini gue lakukan sebenarnya hanya untuk

kemudahan saja dalam mengerti penggunaan APC. Kalau kalian sudah mengerti, silahkan implementasikan dengan cara kalian sendiri-sendiri, dan gunakan Framework apa saja yang kalian suka.

## Upload Form

Mari kita bikin upload form yang sederhana saja.

```
1. <form action="upload_progress.php" enctype="multipart/form-data"
2.   id="upload_form" method="post"
3.   onsubmit="postForm('upload_form')";
4.   return false;" target="hiddenframe">
5.   <input id="progress_key" name="APC_UPLOAD_PROGRESS"
6.     value=<?php echo uniqid()?>" type="hidden">
7.   <input id="test_file" name="test_file" type="file">
8.   <input value="Upload!" type="submit">
9. </form>
10. <div id="progress_win">
11. </div>
12. <iframe name="hiddenframe" style="display: none;"></iframe>
```

So, selesai. Sekarang mari lihat satu-satu. di baris pertama untuk form tag, kita punya atribut action yang mengacu pada dirinya sendiri apabila form tersebut di submit. Atribut enctype, karena kita ingin mengupload file, atribut method seperti biasa, post. Kita memiliki atribut Id yang pada atribut onsubmit di berikan sebagai parameter kepada fungsi javascript postForm. Dan kita memiliki atribut target, yang memiliki nilai "hiddenframe". Nilai ini mengacu pada nama iFrame yang didefinisikan di baris 8(delapan).

Dengan menggunakan atribut target pada form tag, maka apabila form tersebut di *submit*, proses submit tersebut akan dijalankan di Iframe, sehingga halaman dimana form tersebut berada tidak akan di reload. Dengan atribut onsubmit, maka sebelum form tersebut di submit, maka fungsi yang ada di dalam onsubmit tersebut dijalankan. *Return false* sebenarnya menyatakan bahwa setelah menjalankan fungsi tersebut maka gagalkan operasi submit form. Kenapa digagalkan? nanti gue kasih tahu di bagian javascript.

Baris berikutnya kita punya satu hidden input field. Field ini **WAJIB** ada apabila kalian ingin menggunakan APC untuk mendeteksi kemajuan upload file. Atribut Id pada input field ini bisa kalian definisikan sesuka kalian, gue definisikan sebagai progress\_key. Atribut name harus sesuai dengan konfigurasi APC pada apc.rfc1867\_name yang bisa dilihat di phpinfo(). setelah itu kita memiliki atribut value yang gue beri nilai yang unik. Nilai yang unik ini penting untuk orientasi APC ketika memproses kemajuan upload file. Apabila pada waktu bersamaan ada beberapa orang yang mengupload file, katakanlah user A dan B dan kalian memberikan id pada atribut value nya sama, maka APC bisa bingung,

2 progress dari 2 orang berbeda, tapi idnya sama, yang mana punya siapa? ^^

Di baris 6 kita punya div tag. Ini fungsinya sebagai container untuk menampilkan *progress bar* nya. Disini gue gak akan menjabarkan cara membentuk progress bar, karena untuk itu sudah banyak tutorialnya, gue hanya akan menunjukkan total prosesnya dalam persen. Sekarang mari kita lihat javascriptnya.

### Javascript

Karena kita akan bekerja menggunakan AJAX, maka mari kita buat fungsi yang mempersiapkan sebuah *instance* dari HTTP Request Object, supaya nanti tinggal dipakai. Thema AJAX sendiri gak akan gue bahas panjang lebar. Contoh dan Tutorial bisa lihat di salah satu [artikel dalam blog gue](#) atau cari di google.

```
1. function createRequestObject()
2. var requestObject;
3. requestObject = false;
4. if( window.ActiveXObject ){
5.     for( var i = 5; i; i- ){
6.         try{
7.             if( i == 2 ){
8.                 requestObject =
9.                     new ActiveXObject( "Microsoft.XMLHTTP" );
10.            }else{
11.                requestObject =
12.                    new ActiveXObject( "Msxml2.XMLHTTP." + i + ".0" );
13.            }
14.            break;
15.        }catch( excNotLoadable ){
16.            requestObject = false;
17.        }
18.    }
19. }else{
20.     requestObject = new XMLHttpRequest();
21. }
22. return requestObject;
23. }
24.
25. var http = createRequestObject();
```

So, dengan ini kita sudah mempersiapkan sebuah *instance* dari HTTP Request Object. Sekarang kita buat fungsi postForm() yang akan dipanggil ketika html form yang barusan kita buat di submit.

```
1. function postForm(formName){
```

```
2. document.getElementById(formName).submit();
3. setTimeout('updateProgress()', 100);
4. }
```

Isinya mudah saja. HTML form dengan nama yang didefinisikan di submit. setelah itu kita menunggu 100ms sebelum menjalankan fungsi updateProgress(). Kenapa harus menunggu? Karena file yang di upload butuh waktu untuk mencapai server. Apabila kita tidak menunggu, maka APC tidak memiliki informasi apapun untuk dikirimkan kembali ke kita, dalam hal ini APC akan mengirimkan nilai false. Dan itu sangat tidak baik.

Interval 100ms gue gunakan hanya untuk lokal server. Untuk remote server ada baiknya untuk mendefinisikan interval yang lebih besar, satu hingga dua detik (2000ms) karena pengiriman data dari client hingga ke server bisa memakan waktu lama tergantung kecepatan internet.

Sekarang mari kita lihat apa isi di dalam fungsi updateProgress().

```
1. function updateProgress(){
2. progress_key = document.getElementById('progress_key').value;
3. http.open("GET", 'upload_progress.php?progress_key='+progress_key, true);
4. http.onreadystatechange = function () {
5.     if (http.readyState == 4) {
6.         if (http.status == 200) {
7.             the_object = eval("(" + http.responseText + ")");
8.             if(!the_object.done){
9.                 result = Math.round(
10.                     (the_object.current/the_object.total) * 100);
11.                 document.getElementById('progress_win').innerHTML =
12.                     result +'';
13.                 setTimeout("updateProgress()",500);
14.             }else{
15.                 document.getElementById('progress_win').innerHTML = '100%';
16.             }
17.         }
18.     }
19. }
20. http.send(null);
21. }
```

Disini kita lihat bahwa kita mengirimkan GET Request ke server dengan parameter progress\_key yang diambil dari input hidden field yang unik tadi. Server nantinya akan mengirimkan response text berupa [JSON](#). Response text tersebut akan dirubah menjadi javascript object dengan eval(). Lalu kita check

apakah object atribut "done" didefinisikan dalam object. Apabila tidak, maka hitung kemajuannya dalam persen, lalu tampilkan nilainya dalam div container. Setelah itu tunggu beberapa saat, dalam hal ini gue memasang interval 500ms, dan ulangi proses tersebut.

Server akan menambahkan atribut "done" apabila proses upload sudah selesai. Dan apabila ini terjadi, maka angka persen di div container akan ditulis "100%". Mengenai definisi return array oleh APC silahkan baca [manualnya](#) pada bagian RFC1867. Dengan ini javascript kita kelar. Sekarang kita bisa lihat PHP scriptnya.

### PHP script

Disini script kita akan meng-*handle* 2 macam proses, yang pertama adalah upload file itu sendiri, dan yang kedua adalah membaca kemajuan proses upload.

```
1. if($_SERVER['REQUEST_METHOD']=='POST') {  
2.     //implement your upload code here  
3.     //e.g. @move_uploaded_file($filename, $destination)  
4.     exit;  
5. } else if(isset($_GET['progress_key'])) {  
6.     $status = apc_fetch(ini_get('apc.rfc1867_prefix').  
7.                         $_GET['progress_key']);  
8.     echo json_encode($status);  
9.     exit;  
10. }else{  
11.     //you can implement some code to display error here  
12. }
```

Scriptnya mudah saja apabila request method yang diterima adalah post, maka upload filenya. disini kita bisa memasukkan code yang biasa kita implementasikan kalau kita ingin mengupload file.

Apabila request method nya bukan post, maka periksa apakah GET parameter dengan nama "progress\_key" di definisikan, jika ya maka ambil informasi tentang status upload file dengan `apc_fetch()`. Fungsi `ini_get()` disini adalah untuk mengambil prefix yang didefinisikan di konfigurasi apc (bisa dilihat di `phpinfo()`). Setiap file yang di upload dan memiliki hidden field "APC\_UPLOAD\_PROGRESS" pada formnya, APC bisa mengidentifikasi kembali melalui prefix yang didefinisikan di `php.ini` dan value yang didefinisikan di hidden field tersebut.

Fungsi `apc_fetch()` akan mengirimkan array atau boolean false. Apabila dia mengirimkan array, maka kita akan merubah array tersebut menjadi JSON dengan `json_encode()` function dan mengirimkannya kembali ke client dalam bentuk plain text.

Jadi deh. Tinggal di test. Source code yang kumplit bisa kalian dapatkan [disini](#) dan di bawah ini hasilnya, kalian bisa langsung coba ^^.

## 22. Membuat Slideshow Headline Berita Sederhana Dengan JQuery (+Efek Animasi)

Jika berbicara tentang sarana interaktif pada web di era saat ini. Efek slideshow tentunya paling sering digunakan oleh website-website yang mengandung content berupa berita. Slideshow biasa diisi dengan rangkuman-rangkuman berita terkini (Headline) agar pengunjung dapat memantau perkembangan berita terkini dengan sajian efek slideshow yang menarik dan tidak akan cepat membuat pengunjung bosan.

Contoh slideshow yang akan kita bangun pada artikel kali ini adalah seperti gambar berikut:



Kunci dari pembuatan fasilitas ini pada halaman web anda adalah **CSS + JavaScript**. Untuk membuatnya pertama kali buat halaman HTML seperti ini:

```
<div id="divTrigger">
  <a href="javascript:;" onClick="openContent(this, 'div1')"
  id="firstSlide">1</a>
  <a href="javascript:;" onClick="openContent(this, 'div2')">2</a>
  <a href="javascript:;" onClick="openContent(this, 'div3')">3</a>
  <a href="javascript:;" onClick="openContent(this, 'div4')">4</a>
</div>

<div id="divContent">
  <div id="div1">
    <span class="title">Pemandangan Pantai</span>
     Pemandangan pantai memang menakjubkan, dengan berdiam diri sejenak dan melihat pemandangan pantai niscaya kita akan mendapatkan kedamaian jiwa.
  </div>
  <div id="div2">
    <span class="title">Dua Pinguin</span>
     Pinguin adalah spesies makhluk hidup yang langka dan hanya terdapat pada benua tertentu di bumi ini. Oleh karena itu kita penduduk dunia harus melestarikannya.
  </div>
</div>
```

```

<div id="div3">
    <span class="title">Pepohonan Sejuk</span>
     Ini adalah gambar wallpaper
pilihan saya ketika sedang bekerja, dengan
    melihat gambar ini hati terasa menjadi lebih tenang.
</div>
<div id="div4">
    <span class="title">Hujan Sakura</span>
     Menikmati bunga sakura bagi
warga jepang adalah tradisi musiman yang
    menyenangkan pada saat memasuki musim dingin.
</div>
</div>

```

Lalu persiapkan CSS seperti ini:

```

#divTrigger a:link,#divTrigger a:visited{
    text-decoration: none; /* hilangkan garis bawah pada link */
    margin:4px 0px; /* memberi jarak antar link */
    padding: 1px 4px; /* menjadikan link seperti kotak */
    border: 1px solid #CCC;
    background-color: #FFF;
    color: #000;
}
/* link yang dilalui oleh mouse dan element dengan class='selected' akan
mempunyai efek yang sama */
#divTrigger a:hover,#divTrigger a:active,.selected{
    background-color: #00A;
    color: #FFF;
}
#divContent{
    margin-top:4px;
    border:1px solid #CCC;
    width: 300px;
    height: 220px;
    padding: 4px;
}
#divContent div{
    /* semua div di dalam element dengan id='divContent' akan di
sembunyikan sementara */
    display:none
}
.title{
    font-size: 16px;
    font-weight: bold;
    color: #006;
    text-decoration: underline;
    margin-bottom: 2px;
    display:block
}

```

Dengan menggunakan framework JavaScript JQuery kita akan lebih mudah berkreasi dengan efek slideshow. Contoh script-nya seperti ini:

```
<script language="javascript" src="jquery.js"></script>
```

```

<script language="javascript">
    /* parameter untuk menjalankan fungsi openContent ini adalah
       1. Element trigger (link) yang akan membuka content apabila di click
       2. Id dari content yang akan ditampilkan
    */
    /* Update 08 Maret 2009
       -> Slideshow diberikan animasi otomatis berpindah-pindah content
    */
    // siapkan variable timer yang akan menyimpan fungsi animasi
    var timer = null;
    function openContent(trigger,divID){
        // semua link pada div dengan id='divTrigger' akan di setarakan style-nya menjadi style normal
        $('#divTrigger a').each(
            function(){
                $(this).css({'background-color':'#FFF','color':'#000'});
            }
        );
        // semua div di dalam element dengan id='divContent' disembunyikan
        $('#divContent div').hide();

        // div yang akan ditampilkan diberi efek fadeIn (built-in dari JQuery)
        ketika ditampilkan
        $('#'+divID).fadeIn('slow');

        // link menjadi trigger diberi style berbeda dengan link lainnya agar
        dapat diketahui content nomor berapa yang sedang aktif
        $(trigger).css({'background-color':'#00A','color':'#FFF'});

        // Update 08 Maret 2009
        // timer di set
        if(timer != null) clearTimeout(timer);
        timer = setTimeout(
            function(){
                /* Cek terlebih dahulu apakah link yang sedang di-click saat
                ini ada link lagi setelah itu
                   apabila tidak ada link lagi setelah element yang sedang di-
                click maka pilih element link pertama
                   dengan selector ':first' dari jquery. Setelah sudah
                ditentukan link, maka lakukan event click
                   pada link tersebut. Setiap link yang di click akan
                menjalankan fungsi ini sehingga terjadi animasi
                   perpindahan content slideshow.
                */
                var nextAnchor = ($(trigger).next('a').text() == '') ?
                    $('#divTrigger a:first') : $(trigger).next('a');
                nextAnchor.click();
            }, 3000 // 3 detik waktu perpindahan content
        );
    }
    $(document).ready(
        function(){
            // untuk permulaan, tampilkan content nomor pertama
            '#firstSlide' adalah id dari trigger yang akan membuka content nomor pertama
            openContent($('#firstSlide'), 'div1');
        }
    )

```

```
)  
</script>
```

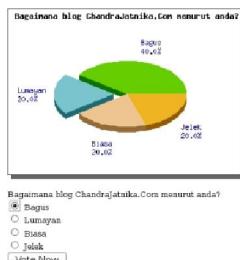
Download juga tutorial agar data slideshow dapat diambil dari database sehingga slideshow menjadi dinamis [\*\*DISINI\*\*](#).

Demo aplikasi dapat anda lihat di [\*\*Link ini\*\*](#)

Download aplikasi lengkap pada [\*\*Link ini\*\*](#)

Semoga bisa bermanfaat... ^\_^

## 23. Polling Dengan Ajax Dan Menampilkan Hasil Polling Dengan Grafik Pie / Oval



Polling atau vote adalah aplikasi yang sering dijumpai pada sebuah website. Fungsinya adalah untuk menjaring pendapat tentang suatu permasalahan atau pertanyaan umum dari para pengunjung website. Pendapat-pendapat tersebut lalu akan dirangkum dalam perhitungan presentase dan ditampilkan dengan berbagai bentuk seperti chart, diagram atau cuma sekedar angka agar para pengunjung tau rangkuman pendapat dari pertanyaan polling tersebut.

Pada artikel kali ini saya akan coba menggabungkan **Ajax** dan **PHP GD** untuk membangun aplikasi polling sederhana. Ajax akan berfungsi untuk memproses pilihan pengunjung dari form dan menyimpannya ke database. PHP GD berfungsi untuk menampilkan grafik dari hasil pilihan pengunjung, grafik yang dipilih kali ini adalah dengan bentuk Oval 3 dimensi. Hasil dari aplikasi polling sederhana ini adalah seperti gambar di bawah ini.

Untuk membuat aplikasi ini ada beberapa library yang dilibatkan yaitu:

- **JQuery + Plugin Ajax Form:** Untuk memproses data polling dan menyimpannya ke database melalui PHP. Ajax form dapat mempermudahkan proses pengiriman data dari form ke file PHP melalui back-end atau tanpa reload. Lebih jelasnya lihat artikel saya [di sini](#).
- **JPGraph:** Adalah library PHP yang berfungsi untuk mempermudah pembuatan berbagai macam gambar grafik, chart dan lainnya. Ingin lebih tahu tentang JPGraph ada di [link ini](#).

Langkah-langkah pembuatannya persiapkan terlebih dahulu table pada MySQL yang akan menyimpan data polling.

```
CREATE TABLE `polling` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `question` varchar(255) collate latin1_general_ci NOT NULL,
  `answer1` varchar(100) collate latin1_general_ci NOT NULL,
  `answer2` varchar(100) collate latin1_general_ci NOT NULL,
  `answer3` varchar(100) collate latin1_general_ci NOT NULL,
  `answer4` varchar(100) collate latin1_general_ci NOT NULL,
  `vote1` int(10) NOT NULL,
  `vote2` int(10) NOT NULL,
  `vote3` int(10) NOT NULL,
  `vote4` int(10) NOT NULL,
  PRIMARY KEY (`id`)
);

-- Insert one row

INSERT INTO `polling` VALUES (NULL, 'Bagaimana blog ChandraJatnika.Com menurut anda?', 'Bagus', 'Lumayan', 'Biasa', 'Jelek', 1, 1, 1, 1);
```

Lalu ambil data tersebut dan tampilkan pada form. Berikut keseluruhan script di dalam tag BODY.

```

<div id="divLoading" style="font-size:10px"></div>
50 <br /><br />
51 <div id="divResult" style="font-size:10px; color: green; font-
52 weight:bold"></div>
53 <form action="vote.php" method="post" id="formVote">
54 <?
55     include 'connect.php'; // koneksi ke database mysql
56
57     $query = mysql_query ('SELECT
58 question,answer1,answer2,answer3,answer4 FROM polling');
59     $data = mysql_fetch_object($query);
60     echo $data->question;
61
62     // tampilkan pilihan dengan format radiobutton (hanya 1 pilihan)
63     echo '<br /><input type=radio name=vote value=1> '.$data->answer1;
64     echo '<br /><input type=radio name=vote value=2> '.$data->answer2;
65     echo '<br /><input type=radio name=vote value=3> '.$data->answer3;
66     echo '<br /><input type=radio name=vote value=4> '.$data->answer4;
67     echo '<br /><input type=submit value="Vote Now">';
68 ?>
</form>
```

Masih di file yang mengandung form diatas, tambahkan script di bawah ini pada tag HEAD.

```

10 <script language="javascript" src="jquery.js"></script>
11 <script language="javascript" src="jquery.form.js"></script>
12 <script language="javascript">
13     function reloadImage () {
14         // tampilkan text 'loading' ketika image sedang dimuat browser
15         $('#divLoading').text('Loading...');

16         /* Cara di bawah ini untuk me-refresh gambar
17             syntax (new Date()).getTime() akan me-regenerate nomor
18 unique
19             agar gambar yang di load selanjutnya selalu ter-update.
20             misal: graphic.php?11575614, graphic.php?12216516, dll */
21         $('#imgPollResult').attr('src', 'graphic.php?' + (new
22 Date()).getTime());
23     }
24     // event-event/setup yang akan dijalankan ketika dokument selesai di load
25     $(document).ready(
26         function () {
27             /* fungsi dari plugin ajaxform jquery yang akan memproses data
28 dari
29                 form menuju file proses melalui backend atau tanpa reload.
30                 untuk artikel lebih jelas ttg ajax form lihat di artikel
31 saya sebelumnya
32                 di http://chandrajatnika.com/2008/11/modernisasi-proses-
33 input-form-anda-dengan-jquery-ajax-form/
34             */
35             $('#formVote').ajaxForm(
36                 function () {
37                     $('#divResult').text('Terima kasih atas
38 pilihan anda');
39                     /* ketika data selesai diproses pada file
40 proses maka reload
```

```

41                                image yang menggambarkan statistik hasil
42 vote */
43                         reloadImage();
44 }
45 );
46 $('#imgPollResult').load(
47     function(){
        /* ketika image selesai di load maka text
'loading' di atas gambar
            langsung dihilangkan */
        $('#divLoading').text('');
    }
);

```

Ketika data pada form dikirim, file yang akan mengolah data tersebut lalu memasukannya pada database MySQL adalah file “**vote.php**”.

```

<?
    include 'connect.php'; // koneksi ke database mysql

    $vote = $_POST['vote']; // menerima hasil pilihan dari form berupa
nilai 1,2,3 atau 4
    $SQL = "UPDATE polling SET vote{$vote} = vote{$vote} + 1"; // syntax
update hasil pilihan (hasil sebelumnya ditambah 1)
    mysql_query($SQL);
?>

```

Bagian paling penting adalah file PHP yang akan me-regenerate gambar grafik yaitu “**graphic.php**”.

```

<?php
    // lebih jelas tentang jpgraph silahkan kunjungi
    http://www.aditus.nu/jpgraph/
    include ("jpgraph/jpgraph.php");
    include ("jpgraph/jpgraph_pie.php");
    include ("jpgraph/jpgraph_pie3d.php");
    include ("connect.php"); // koneksi ke database mysql

    $query = mysql_query('SELECT * FROM polling');
    $data = mysql_fetch_object($query);

    /* siapkan data yang akan membentuk grafik oval
       pada grafik ada 4 data yang disiapkan */
    $dataArray = array($data->vote1,$data->vote2,$data->vote3,$data-
>vote4);

    // tentukan lebar dan tinggi gambar keseluruhan
    $graph = new PieGraph(350,250,"auto");
    $graph->SetShadow(); // memberi efek bayangan pada frame gambar

```

```

    // judul yang tertera pada bagian paling atas gambar adalah isi
pertanyaan polling
    $graph->title->Set($data->question);
    $graph->title->SetFont(FF_FONT1,FS_BOLD);
    $graph->legend->Hide(); // jangan ada legend (penjelasan tiap warna di
grafik oval)

    // setup data grafik dari array data yang sudah dipersiapkan
sebelumnya
    $p1 = new PiePlot3D($dataArray);
    $p1->ExplodeSlice(1); // memberi jarak salah satu belahan oval dari
bentuk oval keseluruhan
    $p1->SetSize(0.35); // untuk mengatur besarnya oval pada gambar
keseluruhan
    /* Method SetLabels untuk mengatur agar text yang menjelaskan besaran
pada tiap belahan
       oval berformat seperti yang kita inginkan, pada contoh kali ini
setiap option jawaban
       akan diikuti oleh persentase hasil pilihan user pada bagian bawah
text */
    $p1->SetLabels(array("${$data->answer1}\n%.1f%%", "${$data-
>answer2}\n%.1f%%", "${$data->answer3}\n%.1f%%", "${$data->answer4}\n%.1f%%"), 1);

    // masukan setup grafik oval ke frame gambar keseluruhan
    $graph->Add($p1);
    $graph->Stroke(); // generate gambar
?>

```

File “**graphic.php**” dapat dipanggil oleh file html manapun dengan menggunakan tag IMG.  
Perhatikan lagi potongan script form.

51 <**img** src="graphic.php" id="imgPollResult" /><**br** /><**br** />

Untuk melihat demo program ini klik [link ini](#)

Untuk mendownload keseluruhan program ini klik [link ini](#)

Semoga bermanfaat... ^\_^

## 24. Membuat Thumbnail Image Dengan PHP dan Image Slideshow Dengan Thickbox JQuery

Salah satu fasilitas pada website adalah gallery image, dimana para pengunjung dapat melihat kumpulan koleksi gambar dari pemilik web. Biasanya gallery juga digunakan untuk dokumentasi suatu acara.

Pada pemasangan gallery di web ada satu persoalan yang cukup mengganggu, yaitu masalah ukuran file pada tiap gambar. Oleh karena itu gambar dengan ukuran besar sebaiknya tidak boleh langsung ditampilkan di halaman depan gallery anda, lebih baik buat dulu ukuran kecilnya.

Nah, masalah membuat ukuran kecil ini yang cukup merepotkan apabila kita tidak tahu tekniknya, ternyata PHP memberi kemudahan untuk kita dalam me-resize image itu dalam ukuran kecil untuk ditampilkan, bukan sekedar me-resize lebar dan tinggi gambar namun juga me-resize ukuran byte file, teknik ini dinamakan **Thumbnail Image**.

Dengan membuat thumbnail terlebih dahulu tentunya pengunjung anda bisa melihat preview gambar anda sebelum melihat ukuran sebenarnya, dan manfaat lain adalah menghemat bandwidth hosting web anda.

Untuk membuat thumbnail pada PHP, anda harus pastikan **PHP GD** ter-install pada paket PHP anda. PHP GD adalah kumpulan fungsi-fungsi pada PHP untuk memanipulasi image. Untuk mengecek apakah PHP GD sudah ter-install pada paket anda lihat saja di PHPINFO() dan temukan info seperti di gambar di bawah ini.

## gd

GD Support	enabled
GD Version	bundled (2.0.34 compatible)
FreeType Support	enabled
FreeType Linkage	with freetype
FreeType Version	2.1.9
GIF Read Support	enabled
GIF Create Support	enabled
JPG Support	enabled
PNG Support	enabled
WBMP Support	enabled
XPM Support	enabled
XBM Support	enabled

## gettext

GetText Support	enabled
-----------------	---------

Mengecek apakah PHP GD sudah ter-install dari PHPINFO()

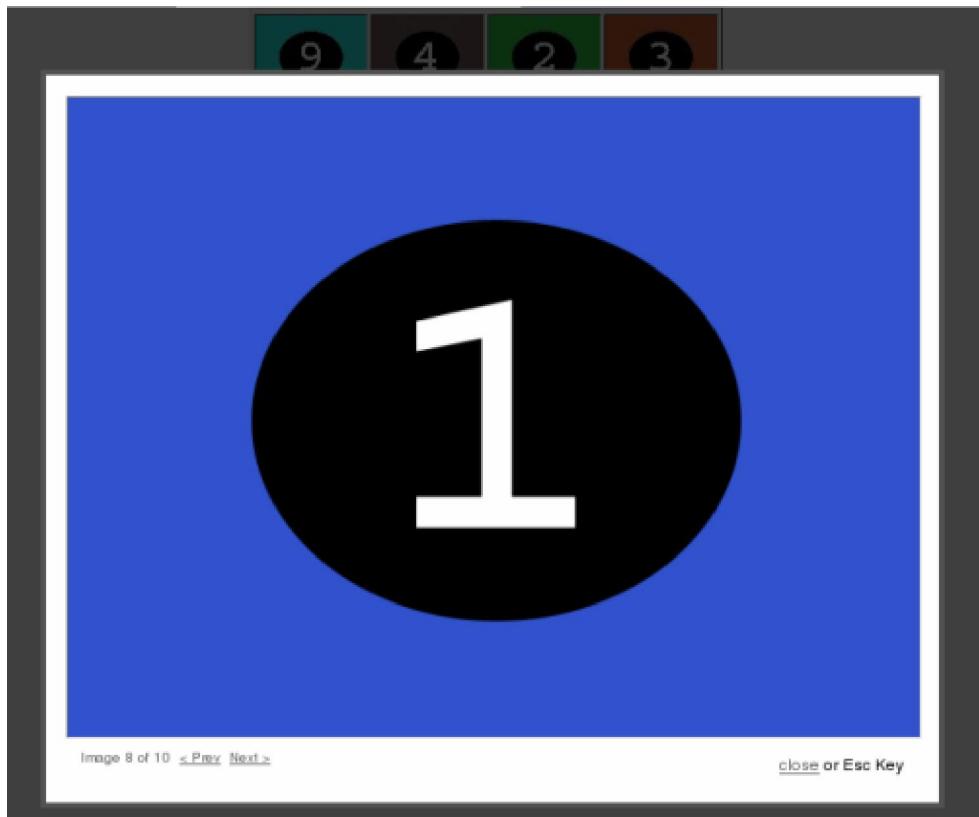
Untuk install PHP GD, mohon tanyakan sama om Google aja yah... keyword-nya "Install PHP GD". ^\_^

Tapi biasanya sih di paket PHP saat ini sudah support PHP GD.

Pada artikel kali ini saya ingin menggabungkan PHP GD untuk membuat thumbnail image dan plugins **thickbox jquery** untuk membuat slideshow ukuran gambar sebenarnya. Untuk lebih jelasnya tentang thickbox, lihat [link ini](#).

Berikut adalah preview hasil dari program:





Langkah-langkah pembuatan program, pertama-tama pada " **index.php** " siapkan script spt ini

```
<html>
<head>
<title>Thumbnail Gallery</title>
<script language="javascript" src="includes/jquery.js"></script>
<script language="javascript" src="includes/thickbox.js"></script>
<link href="includes/thickbox.css" rel="stylesheet" type="text/css" />
</head>
<body>
<div align="center">
<?
$dir = "images/"; // tentukan folder dimana anda menyimpan image
if ($handle = opendir($dir)) {

    echo '<table border=1><tr>';
    $x = 0;
    while (false !== ($file = readdir($handle))) { // looping untuk
mendapatkan daftar file pada suatu folder
        if($file == '.' || $file == '..') continue; // kalau bukan
file gambar dihiraukan
        if($x%4==0) echo '</tr><tr>'; // apabila sudah looping
gambar sebanyak 4x maka buat baris baru
        $path = $dir.$file;
        /*
Perhatikan pada tag <a> ada class="thickbox" --
> agar ketika gambar ini di tekan akan terbuka
```

```

        preview gambar tsb pada layar thickbox.
        Perhatikan juga ada attribute rel="mygallery" -
-> agar dari gambar satu ke gambar lain terhubung
                dan pada layar thickbox terdapat link navigator
untuk slideshow gambar.
                Untuk lebih jelas ttg thickbox silahkan melihat
manual disini
                http://jquery.com/demo/thickbox/
        */
        echo '<td><a href="'.$path.'" class="thickbox"
rel="mygallery"></a></td>';
        $x++;
    }
    echo '</tr></table>';

    closedir($handle);
}
?>
</div>
</body>
</html>

```

Sediakan juga " **thumb.php** " pada direktori yang sama dengan " **index.php** ".

```

<?
    isset($_GET['img']) or die('NO IMAGE');

    include "includes/thumbnail.class.php";
    $thumb = new T10Thumbnail;
    $thumb->setMaxWidth(80); // lebar maksimal untuk setiap image adalah
80px
    $thumb->getThumbnail($_GET['img']); // generate thumbnail image
?>

```

Sedangkan fungsi-fungsi untuk membuat Thumbnail Image ada di file " **thumbnail.class.php** " pada folder "includes/".

```

<?
    class T10Thumbnail{
        var $maxWidth = 200; // default lebar maksimal adalah 200px

        function setMaxWidth($maxWidth){ // method untuk mengubah
nilai lebar maksimal
            $this->maxWidth = $maxWidth;
        }

        function getThumbnail($path){
            $maxheight = $this->maxWidth; // tinggi maksimal
disamakan saja dengan lebar maksimal

            if(file_exists($path)){ // cek apakah file yang akan di
resize ada
                list($width,$height)=getimagesize($path); //
untuk mendapatkan lebar dan tinggi gambar

```

```

        // jika lebar image lebih kecil dari ukuran
        thumbnail yang ditentukan maka tidak perlu di hitung lagi ratio ukurannya
        if($width < $this->maxWidth){
            $new_width = $width;
            $new_height = $height;
        }else{
            if($width>$height) $ratio=$this-
            >maxWidth/$width;
            else $ratio = $maxheight/$height;

            $new_width = $ratio*$width;
            $new_height = $ratio*$height;
        }

        $path_parts = pathinfo($path);
        switch($path_parts['extension']){
            // ambil extension
            file untuk mengecek tipe image
            case "jpeg":
            case "jpg":
                $header = "jpeg";
                $func1="imagecreatefromjpeg"; // fungsi untuk
                generate image jpg
                $func2="imagejpeg"; // fungsi untuk generate
                image jpg
                break;
            case "gif":
                $header = "gif";
                $func1="imagecreatefromgif"; // fungsi untuk
                generate image gif
                $func2="imagegif"; // fungsi untuk generate
                image gif
                break;
            case "png":
                $header = "png";
                $func1="imagecreatefrompng"; // fungsi untuk
                generate image png
                $func2="imagepng"; // fungsi untuk generate
                image png
                break;
        }
        // membuat file php ini ketika dipanggil di browser
        header("Content-type: image/$header");

        $thumb =
imagecreatetruecolor($new_width,$new_height); // buat resource image dalam
ukuran thumbnail
        $source = $func1($path); // generate image

        imagecopyresized($thumb,$source,0,0,0,$new_width,$new_height,$width,$height
        ); // thumbnail ukuran image
        $func2($thumb); // generate image
    }
}
?>

```

Penjelasan program saya langsung tulis di setiap script. Dan untuk melihat demo dari program ini klik [\*\*Link ini\*\*](#).

Download juga program ini di [\*\*Link ini\*\*](#).

Semoga Bermanfaat... Terima kasih... ^\_^

## 25. Membuat Data Pagging Dengan Ajax

Dalam menampilkan data, pagination atau pagging adalah element terpenting karena berfungsi untuk memilah-milah data menjadi beberapa halaman apabila data tersebut dianggap terlalu panjang/banyak untuk ditampilkan pada satu halaman.

Seiring dengan perkembangan web 2.0, kebutuhan akan Ajax dalam setiap halaman web makin digemari dan pada artikel kali ini saya akan memberikan cara agar anda bisa membuat pagging dengan menggunakan Ajax. Oleh karena itu pagging kali ini tidak akan membuat kita menunggu halaman selesai di loading sampai kita bisa tahu isi data pada selanjutnya.

Kali ini saya pakai framework **JQuery** agar script lebih ringan dan mudah dipelajari, langsung saja anda masukan script di bawah ini di tengah-tengah tag HEAD.

```
6 // setup
7 var PHPfile = 'data.php'; // format cara penanganan parameter dan response
8 sudah diatur dalam kesatuan program ini
9 var recordPerPage = 5; // jumlah baris yang ditampilkan tiap halaman
10 var elementTable = '#divTable'; // element yang akan menampilkan data
11 var elementNav = '#divNavigator'; // element yang akan menampilkan link-
12 link pagging
13
14 $(document).ready(
15   function() {
16     $.post(PHPfile+'?action=getdata', // untuk mengambil data masukan
17 alamat "data.php?action=getdata"
18       {limit: recordPerPage, page: 1}, // parameter untuk
19 "data.php?action=getdata" adalah "limit" dan "page", dikirimkan secara post
20       function(data) {
21         if(data != 'EMPTY'){ // jika response tidak berupa kata
22 'EMPTY'
23           $(elementTable).html(data); // masukan hasil
24 response secara langsung ke element yang akan menampilkan data
25
26 $.post(PHPfile+'?action=getcount', function(totalRows){ // untuk mendapatkan
27 total baris keseluruhan pada data yang akan di-pagging gunakan
28 "data.php?action=getcount"
29           if(recordPerPage < totalRows){ // jika total
30 keseluruhan baris lebih besar dari total baris per halaman maka tampilkan
31 pagging
32             var i = Math.ceil(totalRows/recordPerPage);
33 // untuk mendapatkan berapa banyak link pagging yang akan ditampilkan
34             var navigators = []; // siapkan array
35 navigator yang akan menyimpan html link
36
37             for(var j=1;j<=i;j++) // looping angka
38 navigator
39               navigators.push('<a
40 href="javascript:;">' + j + '</a>') // navigator berupa link sederhana, anda
41 bisa custom dengan style anda sendiri
42
43             $(elementNav).html(navigators.join(' '));
44 // hasil array link navigator di masukkan ke element yang akan menampilkan
```

```
45 link pagging
46 $(elementNav+' a').click( // beri event
47 onClick pada setiap link navigator
48         function() {
49             var _page = $(this).text(); // ambil
angka pada link pagging

pada link pagging
                                // ambil lagi data berdasar angka
                                // pada link pagging
                                $.post(PHPfile+'?action=getdata',
                                {limit: recordPerPage, page:
                                _page},
                                function(data) {
                                    if(data != 'EMPTY')
                                        $ (elementTable).html(data);
                                    }
                                );
                                $(elementNav+' a').css('font-
weight','normal'); // semua link pagging di normalkan style-nya
                                $(this).css('font-weight','bold');
// link pagging yang baru saja di click di beri efek bold
                                }
                            );
                            }
                        );
                    });
                }
            );
        }
    );
});
```

Script diatas berfungsi untuk membuat pagging dan menampilkan data pagging pada element-element HTML yang sudah ditentukan. Lalu pada BODY HTML cukup siapkan saja dua tag DIV.

```
53 <div id="divTable" style="height: 160px;"></div>
54 <div id="divNavigator"></div>
```

Element yang tak kalah penting adalah proses data pada PHP, berikut adalah file PHP yang mempunyai 2 fungsi yaitu menampilkan table data dan memberikan response total keseluruhan data yang nantinya akan dijadikan link-link pagging oleh javascript.

```
<?
    isset($_GET['action']) or die('error params'); // file ini harus
dipanggil dengan parameter "action", misal "data.php?action=getdata" atau
"data.php?action=getcount"

    mysql_connect('localhost', 'root', '');
    mysql_select_db('test');

    switch($_GET['action']){
        case 'getdata':
            $page = intval($_POST['page']); // index halaman
```

```

$limit = intval($_POST['limit']); // jumlah baris per pagging
/* ambil batas limit,
 * misal page = 1, jumlah baris = 10
 * maka batas limit = (1-1) * jumlahbaris = 0 -> SELECT *
FROM pegawai LIMIT 0,10
* misal page = 2, jumlah baris = 10
* maka batas limit = (2-1) * jumlahbaris = 10 -> SELECT *
FROM pegawai LIMIT 10,10
* */
$startPage = ($page-1) * $limit;

$sql = "SELECT * FROM pegawai LIMIT $startPage,$limit";
$query = mysql_query($sql);
if(mysql_num_rows($query) > 0) {
?>


| NIP             | Nama             | Umur             | Alamat             |
|-----------------|------------------|------------------|--------------------|
| <?=\$row->NIP?> | <?=\$row->nama?> | <?=\$row->umur?> | <?=\$row->alamat?> |
| <?=\$row->NIP?> | <?=\$row->nama?> | <?=\$row->umur?> | <?=\$row->alamat?> |


<?
} else echo 'EMPTY';
break;
case 'getcount':
$sql = "SELECT count(NIP) FROM pegawai"; // ambil total record pada table pegawai
list($total) = mysql_fetch_row(mysql_query($sql));
echo $total;
break;
}
?>

```

Untuk melihat langsung demo dari program ini dapat anda lihat di [link ini](#)

Untuk mendownload program melalui [link ini](#)

Semoga bermanfaat... Terima kasih... ^\_^

## 26. Membuat Tab Pada Halaman Web Dengan Lebih Mudah

Pada saat ini teknologi web semakin gencar dikembangkan oleh developer agar dapat menyaingi keunggulan pemrograman desktop, hal yang nyata terjadi adalah semakin banyak interface form desktop yang bisa diimplementasikan di halaman web. Contohnya seperti **DateTimePicker**, **ColorPicker**, **DynamicGrid**, **Slider** dan **PageTab**. Khusus untuk PageTab atau Tab saya akan memberikan kemudahan dalam mengimplementasikannya ke aplikasi anda.

Interface Tab sangat berguna untuk memilah-milah halaman atau form, contohnya anda bisa lihat di gambarnya di bawah ini.



Dengan adanya tab, kita bisa memasukan banyak content pada 1 halaman dan merapihkannya dengan metode page tab.

Biasanya untuk membuat tab kita butuh banyak setup seperti CSS, javascript, pengaturan HTML dll. Namun saya akan memberikan kemudahan pada class yang saya buat di bawah ini, berikut adalah script dari class "tabs.class.js". Script ini hanya membutuhkan framework **prototype.js**.

```
var T10Tabs = Class.create(
{
    // method initialize adalah konstruktor dari class T10Tabs
    initialize: function(tabprefix,contentprefix){
        /* tidak melakukan apa-apa, anda bisa menambahkan beberapa
           perintah disini apabila ingin mengembangkan */
    }
})
```

```

},
addTab: function(tabID, tabContentID) {
    this.setNormalStyle(tabID); // Membuat element tabID menjadi
style normal (belum diseleksi)
    $(tabContentID).setStyle({
        padding: '4px',
        border: '1px solid #000',
        backgroundColor: '#B4D4F1',
        clear: 'left', // untuk menghilangkan efek
'float' dari style tab
        display: 'none', // secara default tab content
tidak ditampilkan terlebih dahulu
        height: '200px'
    });
    T10Tabs.tab_id.push(tabID); // simpan element tab ke dalam
array tab_id
    T10Tabs.tab_content.push(tabContentID); // simpan element
content tab ke dalam array tab_content

    var index = T10Tabs.tab_id.length - 1; // ambil index terakhir
dengan cara = panjang array tab dikurang 1
    Event.observe(tabID, 'click', // apabila tab di click maka
tampilkan content-nya
        function() {
            (new T10Tabs()).showTab(index);
        }
    );
},
setNormalStyle: function(tabID){ // ubah menjadi style normal (belum
diseleksi)
    $(tabID).setStyle({
        padding: '4px',
        border: '1px solid #000',
        backgroundColor: '#FFF',
        float: 'left', // diberi nilai float agar
element merata ke kiri (walaupun element tsb adalah div)
        marginLeft: '1px', // memberi jarak antar tab
(anda bisa custom sendiri)
        cursor: 'pointer',
        position: 'relative', // nilai ini harus ada
untuk pengaturan tab selected nantinya
        zIndex: 5, // untuk mengedepankan element tab
agar bisa 'menindih' content tab pada saat diseleksi
        top: '1px' // akan diubah menjadi 2px apabila
tab tsb dipilih (sudah terseleksi)
    });
},
setSelectedStyle: function(tabID){ // ubah menjadi style selected
(sudah terseleksi)
    $(tabID).setStyle({
        top: '2px', // dengan sendirinya tab tsb
'menindih' element content sehingga kelihatan menyatu dengan content tsb
        backgroundColor: '#B4D4F1', // background
disamakan dengan background content agar kelihatan seperti menyatu
        borderBottom: '0px' // dikosongkan border
bottom agar tab dengan content tidak ada garis pemisah dan kelihatan menyatu
    });
}

```

```

},
showTab: function(index) {
    if(T10Tabs.tab_id[index] == undefined) return; // jika tab dengan index yg disebut tidak ada maka tidak mengerjakan apa-apa
    T10Tabs.tab_id.each( // semua tab di jadikan style normal terlebih dahulu
        function(element) {
            (new T10Tabs).setNormalStyle(element);
        }
    );
    T10Tabs.tab_content.each( // semua tab content di sembunyikan terlebih dahulu
        function(element) {
            $(element).hide();
        }
    );
    this.setSelectedStyle(T10Tabs.tab_id[index]); // jadikan tab dengan index yg dikirimkan dengan style selected tab
    $(T10Tabs.tab_content[index]).show(); // tampilkan content tab

    /* simpan tab yang sedang aktif dengan cookie sehingga ketika di refresh dapat kembali ke tab terakhir */
    var currentDate = new Date();
    currentDate.setDate( currentDate.getDate() + 2 ); // set expired date sampai dengan tanggal besok
    document.cookie = this.cookieName+"="+index+"; expires="+currentDate.toGMTString()+"; path=/";
},
getLastTab: function(){ // fungsi untuk mendapatkan tab terakhir
    var results = document.cookie.match( '(^|;) ?' + this.cookieName + '=([^\;]*)(;\|$)' );
    if( results )
        return ( unescape( results[2] ) );
    else
        return 0;
},
cookieName: 'tabCookie' // nama cookie untuk menyimpan tab yang sedang aktif
}
);
T10Tabs.tab_id = []; // untuk menyimpan data tab secara array numeric
T10Tabs.tab_content = []; // untuk menyimpan data content tab secara array numeric

```

Untuk penggunaannya anda tinggal siapkan element-element yang akan dijadikan **TAB** dan **CONTENT TAB** seperti di bawah ini.

```

22 <div id="tab1">Tab 1</div>
23 <div id="tab2">Tab 2</div>
24 <div id="tab3">Tab 3</div>
25 <div id="content1"><h1>CONTENT 1</h1></div>
26 <div id="content2"><h1>CONTENT 2...</h1></div>
27 <div id="content3"><h1>CONTENT 3...</h1></div>

```

Lalu setup tab pada saat dokumen selesai di load.

```
6     window.onload = function() {
7         var tab = new T10Tabs();
8         tab.addTab('tab1', 'content1');
9         tab.addTab('tab2', 'content2');
10        tab.addTab('tab3', 'content3');
11        /* load tab berdasar tab yang aktif terakhir kali sebelum di-
12        refresh           default tab adalah 0 atau tab paling pertama */
13        tab.showTab(tab.getLastTab());
14    }
```

Untuk demo script diatas anda bisa melihat di [link ini](#)

Untuk men-download script keseluruhan klik [link ini](#)

Semoga bisa bermanfaat... ^\_^

## 27. Modernisasi Proses Input Form Anda Dengan JQuery Ajax Form

Dalam menginput data, sisi keamanan untuk memfilter data yang diinput adalah yang paling penting, karena tidak semua user dapat memasukan data dengan benar. Oleh karena itu butuh sebuah **validasi** yang baik sebelum data tersebut dimasukan ke dalam database.

Validasi yang baik menurut pengalaman saya adalah validasi yang disusun secara berlapis baik dari sisi client/user dan dari sisi server, pada prakteknya sebelum data pada form di kirim ke server maka JavaScript akan memfilter data tersebut lalu mengirimkan ke server apabila data itu sudah benar, dan ini yang disebut validasi dari sisi client/user. Lalu pada sisi server pun (misal: PHP) juga menggunakan validasi yang sama seperti JavaScript sebelum akhirnya data masuk ke database.

Ada yang cukup merepotkan dari teori diatas yaitu JavaScript (sisi client) dan PHP (sisi server) harus melakukan validasi yang sama. Misalnya saja dalam memeriksa inputan email, JavaScript dan PHP harus menyiapkan fungsi yang sama untuk memeriksa format alamat email apakah sudah benar atau tidak, cukup merepotkan bukan?

Kalau ditanya validasi mana yang paling ampuh antara sisi client dan server maka jawaban saya adalah sisi server, karena JavaScript bisa saja di non aktifkan dengan bantuan browser sehingga validasi dari sisi client bisa dilewatkan begitu saja. Namun kelemahan dari sisi server adalah user harus rela menunggu loading halaman ketika proses berlangsung.

Solusi dari saya adalah JavaScript disini tidak berfungsi sebagai validasi inputan namun hanya berfungsi sebagai pengantar data ke proses PHP dengan metode AJAX sehingga user tidak harus menunggu loading halaman lalu apabila user non aktifkan JavaScript maka proses tetap berlangsung walau harus loading halaman. Solusi tools untuk melakukan seperti ini adalah **JQuery** dengan bantuan plugin **Ajax Form**.

Pertama-tama siapkan table “**guestbook**” pada database anda (salin script di bawah):

```
CREATE TABLE `guestbook` (
`id` INT( 5 ) NOT NULL AUTO_INCREMENT PRIMARY KEY ,
`name` VARCHAR( 30 ) NOT NULL ,
`email` VARCHAR( 30 ) NOT NULL ,
`message` VARCHAR( 255 ) NOT NULL ,
`submitdate` DATETIME NOT NULL
)
```

Buat form sederhana dengan script seperti ini.

```
23 <form name="formGuestBook" action="proses.php" method="post"
24 onsubmit="return submitForm(this);">
25 <table border="0" cellspacing="0" cellpadding="0">
26   <tr>
27     <td>Name</td>
28     <td>:</td>
```

```

29      <td><input name="name" type="text" size="30" /></td>
30  </tr>
31  <tr>
32      <td>Email</td>
33      <td>:</td>
34      <td><input name="email" type="text" size="30" /></td>
35  </tr>
36  <tr valign="top">
37      <td>Pesan</td>
38      <td>:</td>
39      <td><textarea name="message" cols="40" rows="3"></textarea></td>
40  </tr>
41  <tr valign="top">
42      <td>&ampnbsp</td>
43      <td>&ampnbsp</td>
44      <td><input type="submit" name="submit" value="Submit" />
45          <input type="reset" value="Cancel" /></td>
46  </tr>
47 </table>
</form>

```

Pada tab **HEAD** di tambahkan script ini

```

function submitForm(form) {
9     $(form).ajaxSubmit({
10         success: function(response){ // jika proses pada server selesai
11             if(response.indexOf('Terima kasih') > -1){ // jika response
12                 mengandung kata "Terima kasih"
13                 alert(response);
14                 $(form).clearForm(); // mengosongkan semua inputan
15             }else alert('Error: '+response);
16         }
17     }); // end ajax submit
18     return false; // untuk mencegah form mengirim langsung data ke
19 proses.php
}

```

Lalu siapkan script PHP dengan nama “**proses.php**“.

```

<?
extract($_POST);
if(!empty($name) && !empty($email) && !empty($message)) {
    if(valid_email($email)){
        mysql_connect('localhost','root','');
        mysql_select_db('test');
        $SQL = "INSERT INTO guestbook
VALUES ('','$name','$email','$message',now())";
        if(mysql_query($SQL)) echo 'Terima kasih atas pesan anda';
        else echo 'Maaf, data gagal diinput';
    }else echo 'Email Tidak Valid';
}else echo 'Lengkapi Form';

function valid_email($address) {

```

```
        return ( ! preg_match("/^([a-z0-9+_]+)(.[a-z0-9+_]+)*@([a-  
z0-9-]+\.)+[a-z]{2,6}$/ix", $address) ) ? FALSE : TRUE;  
    }  
?>
```

Fungsi **ajaxSubmit** bawaan plugin Ajax Form dengan otomatis mengirim inputan dari form ke proses.php seperti layaknya submit biasa namun kali ini tanpa ada loading halaman. Dengan kemudahan seperti itu maka anda tidak harus repot mempersiapkan data sebelum dikirim ke server.

Baca juga manual dari Ajax Form [disini](#).

Untuk melihat demo langsung dari script diatas [disini](#).

Untuk download script diatas [disini](#).

Semoga artikel ini dapat membantu dan menambah wawasan... Terima kasih... ^\_^

## 28. Mematikan Interface Dan Menampilkan Alert Dengan Pilihan Design Sendiri

JavaScript bukan hanya dijadikan pelengkap dalam interface web, namun juga menjadi bagian penting dari proses suatu transaksi dalam web.

Pada kesempatan kali ini saya ingin membeberkan cara agar JavaScript dapat mematikan interface yang sedang aktif dan menampilkan alert message dengan design keinginan kita sendiri. Contoh yang saya sajikan dapat anda coba tanpa ada bantuan framework JavaScript apapun.

Pertama-tama kita deklarasikan CSS sederhana seperti ini:

```
1 #divLock{  
2     position:absolute;  
3     top:0px;  
4     left:0px;  
5     background-color:#000;  
6     opacity:0.7;  
7     visibility:hidden;  
8     z-index:10;  
9 }  
10.window{  
11     position:absolute;  
12     top:0px;  
13     left:0px;  
14     border:1px solid #000;  
15     background-color:#CCCCCC;  
16     visibility:hidden;  
17     z-index:11;  
18     width:400px  
19 }
```

Pada script diatas kita sediakan element dengan ID *divLock* yang akan menjadi layer yang akan mematikan interface sementara, maksudnya adalah layer ini yang nantinya menutupi keseluruhan layar menjadi hitam dan user tidak akan bisa mengoperasikan apapun ketika layar ditutup.

Langkah kedua adalah inti dari trik ini yaitu JavaScript:

```
1 function DivWindow () {  
2     var div = document.getElementById ('divLock')  
3     if (div == null) {  
4         document.body.innerHTML += '<div id="divLock"></div>';  
5         this._div = document.getElementById ('divLock')  
6     }  
7     this.openWindow = _openWin;  
8     this.closeWindow = _closeWin;  
9 }  
10function _openWin (winName) {  
11     var win = document.getElementById (winName)  
12     if (win != null) {
```

```

13     with(this._div.style) {
14         visibility = 'visible';
15         width = document.body.clientWidth + 'px';
16         height = screen.height + 'px';
17     }
18     with(win.style) {
19         visibility = 'visible';
20         left = ((document.body.clientWidth/2) -
21 (win.offsetWidth/2)) + 'px';
22         top = '200px';
23     }
24 }
25 }
26 function _closeWin(winName) {
27     var win = document.getElementById(winName)
28     if(win != null) {
29         win.style.visibility = 'hidden';
30         this._div.style.visibility = 'hidden';
31     }
32 }
33
34 window.onload = function() {
35     var div = new DivWindow();
36     document.getElementById('btnOpen').onclick = function() {
37         div.openWindow('divAlert');
38     }
39     document.getElementById('btnClose').onclick = function() {
40         div.closeWindow('divAlert');
41     }
}

```

Script diatas saya buat dengan konsep **OOP** yang dapat anda lihat di [artikel sebelumnya](#). Selanjutnya adalah implementasi pada HTML, yaitu:

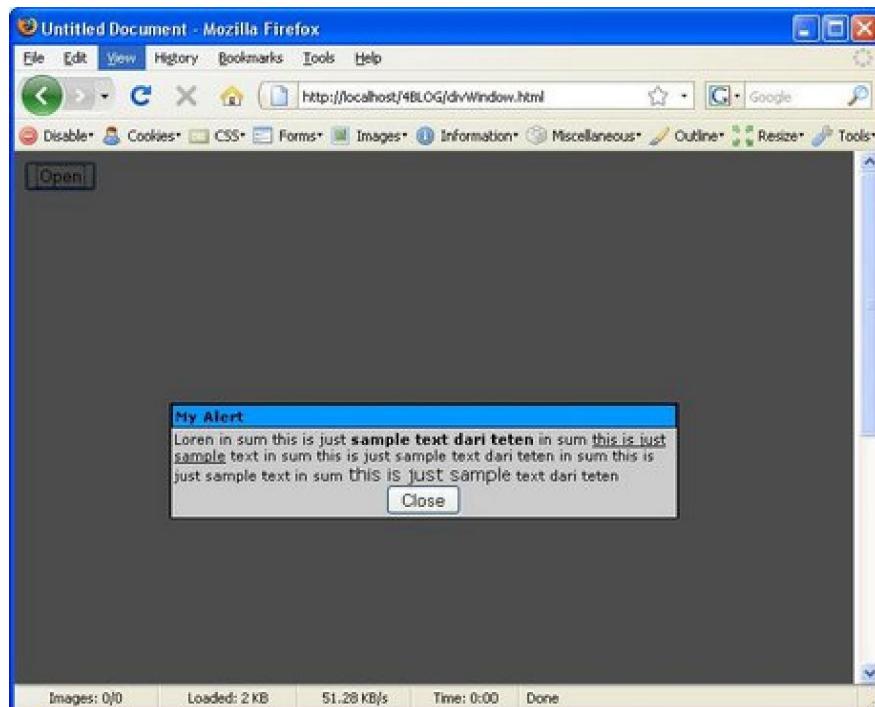
```

<div id="divAlert" class="window">
    <div style="font-weight:bold; padding:2px; border:1px solid #000;
background-color:#0099FF">
        My Alert
    </div>
    <div style="padding:2px;">
        Loren in sum this is just <b>sample text dari teten</b>
        in sum <u>this is just sample</u> text in sum this is just sample
text dari teten
        in sum this is just sample text in sum <big>this is just
sample</big> text dari teten
        <center><input type="button" value="Close" id="btnClose" /></center>
    </div>
</div>
<input type="button" value="Open" id="btnOpen" />

```

Pada interface diatas saya sediakan sebuah tombol yang akan mengawali fungsi untuk mematikan layar yaitu tombol *btnOpen*, dimana deklarasi event-nya saya sudah lakukan pada listing javascript sebelumnya pada baris ke 35 s.d 37.

Maksud alert dengan design sendiri pada listing html diatas adalah pada div dengan ID *divAlert* dimana dia memakai class *window* yang memiliki properti *visibility = "hidden"*. *divAlert* ini disembunyikan sementara sampai dengan ditekannya tombol *btnOpen* yang akan menampilkan layer ini di tengah – tengah halaman setelah terlebih dahulu mengaktifkan layer yang menutupi interface. Gambar di bawah ini dapat memperjelas penjelasan saya sebelumnya.



Ketika layer hitam menutupi interface, user tidak dapat melakukan apa-apa kecuali menutup alert yang ada di hadapan mereka. Inilah gambaran trik alert design sendiri, anda dapat mengubah design alert ini sesuka hati.

Semoga bermanfaat... ^\_^

## 29. Membuat Web Service dengan PHP

### Membuat Server

Untuk membuat server yang menyediakan web service, kita harus membuat sebuah kelas yang berisi method-method yang akan kitajadikan web service.

```
class OperasiMatematika {
```

```
    function tambah($a, $b) {  
        return $a + $b;  
    }
```

```
    function kurang($a, $b) {  
        return $a - $b;  
    }
```

```
    function kali($a, $b) {  
        return $a * $b;  
    }
```

```
    function bagi($a, $b) {  
        if ($b == 0) {  
            /* error */  
        }  
        else {  
            return $a / $b;  
        }  
    }
```

```
    function mod($a, $b) {  
        return $a % $b;  
    }
```

```
}
```

Kelas tersebut akan menjadi dasar web service yang akan kita buat. Setelah itu, kita harus membuat daftar method-method yang akan dijadikan web service. Agar pendefinisian kelas dan pendefinisian daftar method untuk web service tidak dicampur, kita akan membuat daftar method-method tersebut dalam kelas turunannya saja.

```
class OperasiMatematikaService extends OperasiMatematika {
```

```
    var $__dispatch_map;
```

```
    function OperasiMatematikaService() {
```

```

$this->__dispatch_map['tambah'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'int')
);

$this->__dispatch_map['kurang'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'int')
);

$this->__dispatch_map['kali'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'int')
);

$this->__dispatch_map['bagi'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'float')
);

$this->__dispatch_map['mod'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'int')
);

}

function __dispatch($method_name) {

if (isset($this->__dispatch_map[$method_name])) {
    return $this->__dispatch_map[$method_name];
}
else {
    return NULL;
}
}

```

Perhatikan potongan source code di bawah ini.

```

$this->__dispatch_map['mod'] = array(
    'in' => array('a' => 'int', 'b' => 'int'),
    'out' => array('hasil' => 'int')
);

```

Bagian tersebut berisi penjelasan method yang didaftarkan untuk menjadi web service. Nilai dari key dari variabel `$this->_dispatch_map`, dalam hal ini adalah mod, adalah nama method yang akan dijadikan web service. Method ini harus ada di dalam kelas.

Nilai dari variabel tadi menjelaskan mengenai method yang bersangkutan. Dalam contoh di atas, method mod adalah method yang memiliki tipe operasi Request-response, yaitu method yang menerima request dan akan mengeluarkan response.

Jika kita lihat pada potongan source code di atas, method mod akan menerima input berupa dua buah variabel, yaitu a dan b yang masing-masing memiliki tipe int. Method tersebut akan mengeluarkan output berupa sebuah variabel hasil dengan tipe int.

Dalam contoh kelas OperasiMatematika, pada method bagi, ada bagian yang baru saya beri tanda `/* error */`. Bagian tersebut akan berisi potongan source code yang dapat mengeluarkan informasi error kepada client dari web service. Potongan source code yang akan kita isikan pada bagian tersebut adalah sebagai berikut.

```
return new SOAP_Fault('Anda melakukan pembagian dengan nol');
```

Method bagi dalam kelas OperasiMatematika akan menjadi sebagai berikut.

```
function bagi($a, $b) {
if ($b == 0) {
return new SOAP_Fault('Anda melakukan pembagian dengan nol');
}
else {
return $a / $b;
}
}
```

Sampai saat ini, Anda telah membuat sebuah deskripsi dari web service yang akan dibuat. Agar kelas-kelas tadi dapat digunakan sebagai web service, kita akan menggunakan fasilitas dari paket SOAP dari PEAR.

```
<?
```

```
include_once('SOAP/Server.php');
include_once('SOAP/Disco.php');

class OperasiMatematika {

}

class OperasiMatematikaService extends OperasiMatematika {

}

$soap = new SOAP_Server();
$soap->_auto_translation = true;
```

```

$matematika = new OperasiMatematikaService();
$soap->addObjectMap($matematika, 'http://schemas.xmlsoap.org/wsdl/');

if(isset($_SERVER['REQUEST_METHOD']) && ($_SERVER['REQUEST_METHOD'] ==
'POST')) {
$soap->service($HTTP_RAW_POST_DATA);
}
else {
header('Content-type: text/xml');
$disco = new SOAP_DISCO_Server($soap, 'OperasiMatematikaService');

if(isset($_SERVER['QUERY_STRING']) && (strcasecmp($_SERVER['QUERY_STRING'],
'wsdl') == 0)) {
print($disco->getWSDL());
}
else {
print($disco->getDISCO());
}
}

?>

```

Lengkapi isi dari kelas OperasiMatematika dan OperasiMatematikaService pada potongan source code di atas agar menjadi source code utuh yang dapat digunakan.

File PHP yang kita buat di atas berisi pendefinisian kelas yang akan menjadi web service, sebuah fasilitas untuk memproses request yang datang, dan fasilitas untuk melihat WSDL atau deskripsi dari web service yang kita buat.

Setelah semua ini dibuat, web service yang Anda buat sudah siap untuk digunakan. Anda dapat menggunakan fasilitas dari Microsoft .NET, Apache AXIS, atau client web service lainnya (seperti yang akan kita buat juga) untuk mengakses web service ini.

Source code

Demo server

WSDL

### Membuat Client

Pembuatan client web service jauh lebih mudah daripada pembuatan server. Yang harus Anda lakukan pertama kali adalah menentukan alamat web service yang akan di akses. Saya akan menggunakan web service yang terletak pada <http://fajran.net/web/arsip/tutorial/web-service-dengan-php/server.php?wsdl> sebagai bahan uji coba pembuatan client web service.

Pertama-tama, Anda harus membuat proxy (client stub) sebagai perantara antara aplikasi client dan web service yang ingin Anda akses.

```
$soap = new SOAP_WSDL('http://fajran.net/web/arsip/tutorial/web-service-dengan-
```

```
php/server.php?wsdl');
$proxy = $soap->getProxy();
```

Setelah proxy berhasil dibuat, Anda dapat menggunakan object \$proxy tadi untuk mengakses method-method yang dijadikan web service seolah-olah method-method tersebut adalah method lokal.

```
$a = 2;
$b = 2;
```

```
$hasil = $proxy->tambah($a, $b);
print("<p>$a + $b = $hasil</p>");
```

```
$hasil = $proxy->kurang($a, $b);
print("<p>$a - $b = $hasil</p>");
```

```
$hasil = $proxy->kali($a, $b);
print("<p>$a * $b = $hasil</p>");
```

```
$hasil = $proxy->bagi($a, $b);
print("<p>$a / $b = $hasil</p>");
```

```
$hasil = $proxy->mod($a, $b);
print("<p>$a % $b = $hasil</p>");
```

Jika script di atas dijalankan, maka hasil yang akan Anda dapatkan adalah sebagai berikut.  
 $2 + 2 = 4$

$$2 - 2 = 0$$

$$2 * 2 = 4$$

$$2 / 2 = 1$$

$$2 \% 2 = 0$$

## 30. Directory Trees With PHP And jQuery

Knowing where your files are is an important task. Every person who uses a computer usually likes to store their files in an organized manner.

A simple way to keep track of many files is to use a directory tree. A directory tree lists out files and directories so that it's easy to find what you're looking for.

In this tutorial, we will create a directory tree using PHP and jQuery. This tree will use the code found in my previous article: [jQuery drop down menus](#).

If you haven't yet read the previous article, please do so. Everything used in that tutorial will be used here.

[View a Demo](#) [Download the Files](#)

As a quick refresher, take a look at the HTML code for the jQuery drop down menu:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>jQuery Drop Down Menu</title>

<!-- CSS For The Menu -->
<link rel="stylesheet" href="style.css" />

</head>
<body>

<!-- Menu Start -->
<div id="jQ-menu">

    <!-- Unordered List With Menu Goes Here -->

</div>
<!-- End Menu -->

<!-- Add jQuery From the Google AJAX Libraries -->
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.2.6/jquery.min.js"></script>

<!-- jQuery Color Plugin -->
<script type="text/javascript" src="jquery.color.js"></script>

<!-- Import The jQuery Script -->
<script type="text/javascript" src="jMenu.js"></script>
```

```
</body>
</html>
```

We will use all three files - index.html, jMenu.js, and style.css - for our directory tree. Remember to rename index.html to index.php. The PHP that we create will go inside the `#jq-menu` divider.

In order to create a directory tree, we can use recursion. To begin, let's layout our code:

```
function createDir($path = './')
{
}

function printFile($file, $path)
{
}

function printSubDir($dir, $path)
{
}

createDir();
```

We will have three functions:

- `createDir($path)` takes in a path (or defaults to the directory which the file is in) to the directory that it is making a list of. It uses both `printFile($file, $path)` and `printSubDir($dir, $path)`.
- `printFile($file, $path)` prints out a link in a `<li>` element using a given filename and path.
- `printSubDir($dir, $path)` takes in a directory and creates a sub-directory by calling `createDir($path)`. This is where the recursion of `createDir($path)` takes place.

We can start off with `printFile($file, $path)`.

```
function printFile($file, $path)
{
    echo "<li><a href=\"\".$path.$file.\"$file</a></li>";
}
```

This method is straightforward. It creates a list element with a link to the given file.

Let's move on to the `printSubDir($dir, $path)` function.

```
function printSubDir($dir, $path)
{
    echo "<li><span class=\"toggle\">$dir</span>";
    createDir($path.$dir."/");
```

```

        echo "</li>";
    }
}

```

This code is the most important part. It uses recursion so that our directory tree prints out all sub-directories.

- Line one prints a `.toggle` span element that shows/hides the sub-directory on click. The format used here is what the jQuery drop down menu uses to create its effects.
- This second statement is where the recursion takes place. A new call to `createDir($path)` will take care of the elements in this sub-directory.
- The last line prints an ending list element to finish off the sub-directory.

Now we can code the `createDir($path)` function.

We must first open up the directory to read out of.

```

function createDir($path = './')
{
    if ($handle = opendir($path))
    {

    }
}

```

The above code also makes sure the directory is valid.

Now we can print out the start and end of an unordered list that will represent this directory:

```

function createDir($path = './')
{
    if ($handle = opendir($path))
    {
        echo "<ul>";
        echo "</ul>";
    }
}

```

At this point, we can read in the filenames:

```

function createDir($path = './')
{
    if ($handle = opendir($path))
    {
        echo "<ul>";
        while (false !== ($file = readdir($handle)))
        {
        }
        echo "</ul>";
    }
}

```

```

    }
}

```

Our `$file` variable holds the filename of the current file. If this filename is actually a directory, we must call the `printSubDir($dir, $path)` function. On the other hand, if it is just a file, we should call the `printFile($file, $path)` function:

```

function createDir($path = './')
{
    if ($handle = opendir($path))
    {
        echo "<ul>";

        while (false !== ($file = readdir($handle)))
        {
            if (is_dir($path.$file) && $file != '.' && $file != '..')
                printSubDir($file, $path);
            else if ($file != '.' && $file != '..')
                printFile($file, $path);
        }

        echo "</ul>";
    }
}

```

Our directory tree is now functional! Unfortunately, it looks a bit odd when sub-directories are placed after files in the list. We can fix this by creating an array which stores the files. Once we have printed out all the directories, we can move on to printing the files:

```

function createDir($path = './')
{
    if ($handle = opendir($path))
    {
        echo "<ul>";

        $queue = array();
        while (false !== ($file = readdir($handle)))
        {
            if (is_dir($path.$file) && $file != '.' && $file != '..')
                printSubDir($file, $path, $queue);
            else if ($file != '.' && $file != '..')
                $queue[] = $file;
        }

        printQueue($queue, $path);
        echo "</ul>";
    }
}

function printQueue($queue, $path)
{
    foreach ($queue as $file)

```

```
    {
        printFile($file, $path);
    }
}

function printFile($file, $path)
{
    echo "<li><a href=\"\".$path.$file.\"$>$file</a></li>";
}

function printSubDir($dir, $path)
{
    echo "<li><span class=\"toggle\">$dir</span>";
    createDir($path.$dir "/");
    echo "</li>";
}

createDir();
```

And that's it! If you would like to access a different directory than the one your file is in, specify the `$path` parameter for the `createDir($path)` function.

## 31. Create a Photo Admin Site Using PHP and jQuery

I'm pleased to present you with part one of a two part series on creating a photo site using PHP, jQuery, and AJAX. Originally, I intended to fit the entire tutorial into one screencast, but that quickly became a pipe dream as I realized that there was simply too much to cover. Nevertheless, even if you only watch this first video, you should learn a great deal.

We'll be retrieving images from a database, creating a simple login form with authentication, and will then allow for our database to be asynchronously updated. Sounds good? If so, let's get into it.



### OUR GOAL

For this lesson, our goal is to create a photo admin section where an administrator can retrieve photos from a database and update the title of his photos simply by clicking on the appropriate text. To make this page more secure, we'll create a simple membership system as well. In subsequent tutorials, we'll expand upon these concepts.

### *Creating Our Tables With MySQL*

Open PHPMyAdmin and create a database called "db". Within this database, we'll need to add two tables: "photos" and "users".

Table	Action	Records	Type	Collation	Size	Overhead
photos		~13	InnoDB	latin1_swedish_ci	16.0 KiB	-
users		~1	InnoDB	latin1_swedish_ci	16.0 KiB	-
2 table(s)	Sum	~14	InnoDB	latin1_swedish_ci	32.0 KiB	0 B

*photos*

Rather than hard-coding our images into our document, we'll store them in a database. Though we could make this table as sophisticated as we want, we'll just keep it simple for now.

Server: localhost > Database: db > Table: photos "InnoDB free: 4096 kB"

Browse		Structure		SQL		Search		Insert		Export		Import		Operations		Empty		Drop	
Field	Type	Collation		Attributes		Null	Default	Extra						Action					
<input type="checkbox"/> id	int(11)					No		auto_increment											
<input type="checkbox"/> title	varchar(300)	latin1_swedish_ci				No													
<input type="checkbox"/> src	varchar(300)	latin1_swedish_ci				No													

Check All / Uncheck All With selected:

---

Print view Relation view Propose table structure

Add 1 field(s) At End of Table At Beginning of Table After id Go

---

Indexes:					Space usage		
Keyname	Type	Cardinality	Action	Field	Type	Usage	
PRIMARY	PRIMARY	13			id	Data	16,384 B
Create an index on 1 columns					Index	0	B
					Total	16,384	B

- id - INT - Primary Key - Auto Increment
  - title - VARCHAR(500)
  - src - VARCHAR(200)

## users

In order to edit our photos, we need to create a simple membership system that will allow our user to login and edit his photos. We obviously don't want to give everybody that right!

Server: localhost > Database: db > Table: users "InnoDB free: 4096 kB"

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<b>id</b>	int(11)			No		auto_increment	
<input type="checkbox"/>	<b>first_name</b>	varchar(100)	latin1_swedish_ci		No			
<input type="checkbox"/>	<b>last_name</b>	varchar(100)	latin1_swedish_ci		No			
<input type="checkbox"/>	<b>email_address</b>	varchar(100)	latin1_swedish_ci		No			
<input type="checkbox"/>	<b>user_name</b>	varchar(40)	latin1_swedish_ci		No			
<input type="checkbox"/>	<b>password</b>	varchar(25)	latin1_swedish_ci		No			

Check All / Uncheck All With selected:

---

Add 1 field(s)  At End of Table  At Beginning of Table  After id Go

---

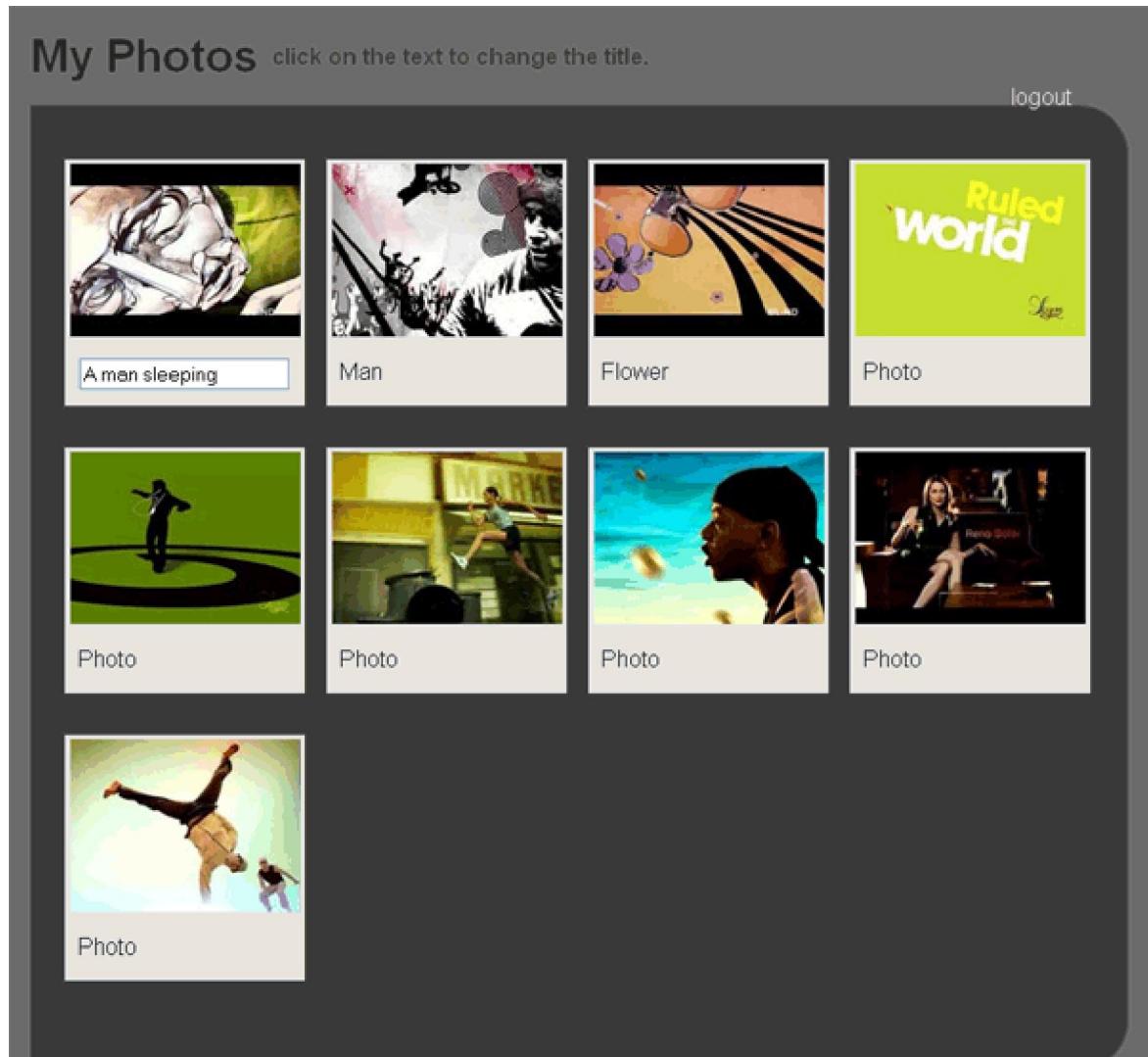
Indexes:					Space usage	
Keyname	Type	Cardinality	Action	Field	Type	Usage
PRIMARY	PRIMARY	1		id	Data	16,384 B
Create an index on 1 columns  Go					Index	0 B
					Total	16,384 B

- id - INT - Primary Key - Auto Increment
  - first\_name - VARCHAR(50)
  - last\_name - VARCHAR(50)

- email\_address - VARCHAR(75)
- user\_name - VARCHAR(25)
- password - VARCHAR(30)

### *Creating the Simple Layout*

---



Create a new php document and add the following into your document.

1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
- 2.
3. <html xmlns="http://www.w3.org/1999/xhtml">
4. <head>
5. <title>My Photos Site

```

6.   <link rel="stylesheet" href="css/default.css" />
7.   <link rel="stylesheet" href="css/jquery.lightbox-0.5.css" />
8.
9.   <script src="js/jquery-1.2.6.pack.js" type="text/javascript">
10.  <script src="js/jquery.lightbox-0.5.pack.js" type="text/javascript">
11.  <script src="js/scripts.js" type="text/javascript">
12. </head>
13. <body>
14. <?php require_once 'photo.php'; ?>
15.
16. <form action="changePhotoTitle.php" method="post">
17.
18. <div id="container">
19.   <h1>My Photos <small>click on the text to change the title.</small></h1>
20.   <a href="login.php?logout=1" id="logout">logout
21.
22.   <div id="main">
23.
24.   <?php require 'getPhotos.php'; ?>
25.
26.   <div id="response" class="hidden" />
27.   </div><!-- end main-->
28. </div><!-- end container -->
29.
30. </form>
31. </body>
32. </html>

```

Don't worry too much about some of what's here. We'll go over everything in time. But as a quick overview, we're importing our jQuery library, a jQuery lightbox plugin, and our custom scripts (that we'll create soon). Take a few seconds to [review the CSS file](#). I won't go over it too much in this written article - as it's long enough already! Refer to the screencast for more information. But, I assure you it's pretty standard and simple stuff.

The primary thing that you need to focus on right now is:

1. <?php require 'getPhotos.php'; ?>

This will be the section where we retrieve our photos from the database. Let's work on that right now. Create a new page called "getPhotos.php".

Let's quickly go over what we need to accomplish with this code.

- Create a connection to the database
- Retrieve all of the rows from the photos table
- If the connection was successful, create an unordered list.
- Filter through the rows and place the necessary information within list item tags.

Paste the following code into your document:

```
1. <?php
2.
3. require 'database.php';
4.
5. $query = "SELECT id, title, src FROM photos";
6.
7. $result = $mysqli->query($query) or die(mysql_error($mysqli));
8.
9. if ($result) {
10. echo "<ul id='photos'> \n";
11.
12. while ($row = $result->fetch_object()) {
13.
14.     $title = $row->title;
15.     $src = $row->src;
16.     $id = $row->id;
17.
18.     echo "<li><a title='$title' href='images/$src'><img src='images/$src' id='$id' alt='$title' /> </a>
\ n";
19.     echo "<h4>$title</h4> \ n";
20.     echo "<input type='text' name='title' value='$title' /></li> \ n \ n";
21.
22. }
23. echo "\ n</ul>";
24. }
25. ?>
```

The first thing we need to do is create a connection to our database. However, we'll be doing this many times over the course of this project. So let's create another page called "database.php" and store the information there. That way, we only need to edit one file, should the need arise. Add the following code into your database.php file.

```
1. <?php
2.
3. $db_name = "users";
4. $db_server = "localhost";
5. $db_user = 'root';
6. $db_pass = "";
7.
8. $mysqli = new MySQLi($db_server, $db_user, $db_pass, $db_name) or die(mysql_error());
9. ?>
```

We're creating a variable called "\$mysqli" and are setting it to a new instance of the "MySQLi" object. We need to pass in four parameters:

- database name
- the server
- username
- password

To keep things clean, we'll place those values in variables and reference them. If the connection isn't made, we'll tell it to "die".

### *Querying the Database*

---

Now let's move back to our "getPhotos.php" file. The next step, is to query the database.

1. \$query = "SELECT id, title, src FROM photos";
2. \$result = \$mysqli->query(\$query) or die(mysql\_error(\$mysqli));

Our query is extremely simple. We're essentially retrieving everything from the database. You might want to modify this query to fit your own application. We pass in the query by using "\$mysqli->query(\$query)".

If that query was performed successfully, we'll create our unordered list and add our images.

1. echo "<ul id='photos'> \n";
- 2.
3. while (\$row = \$result->fetch\_object()) {
- 4.
5. \$title = \$row->title;
6. \$src = \$row->src;
7. \$id = \$row->id;
- 8.
9. echo "<li><a title='\$title' href='images/\$src'><img src='images/\$src' id='\$id' alt='\$title' /> </a>\n";
10. echo "<h4>\$title</h4> \n";
11. echo "<input type='text' name='title' value='\$title' /></li> \n \n";
- 12.
13. }
14. echo "\n</ul>";

We create a variable called \$row that will be equal to each instance of a row in the database. As long as there are rows to be retrieved, this variable will contain that specific row's information.

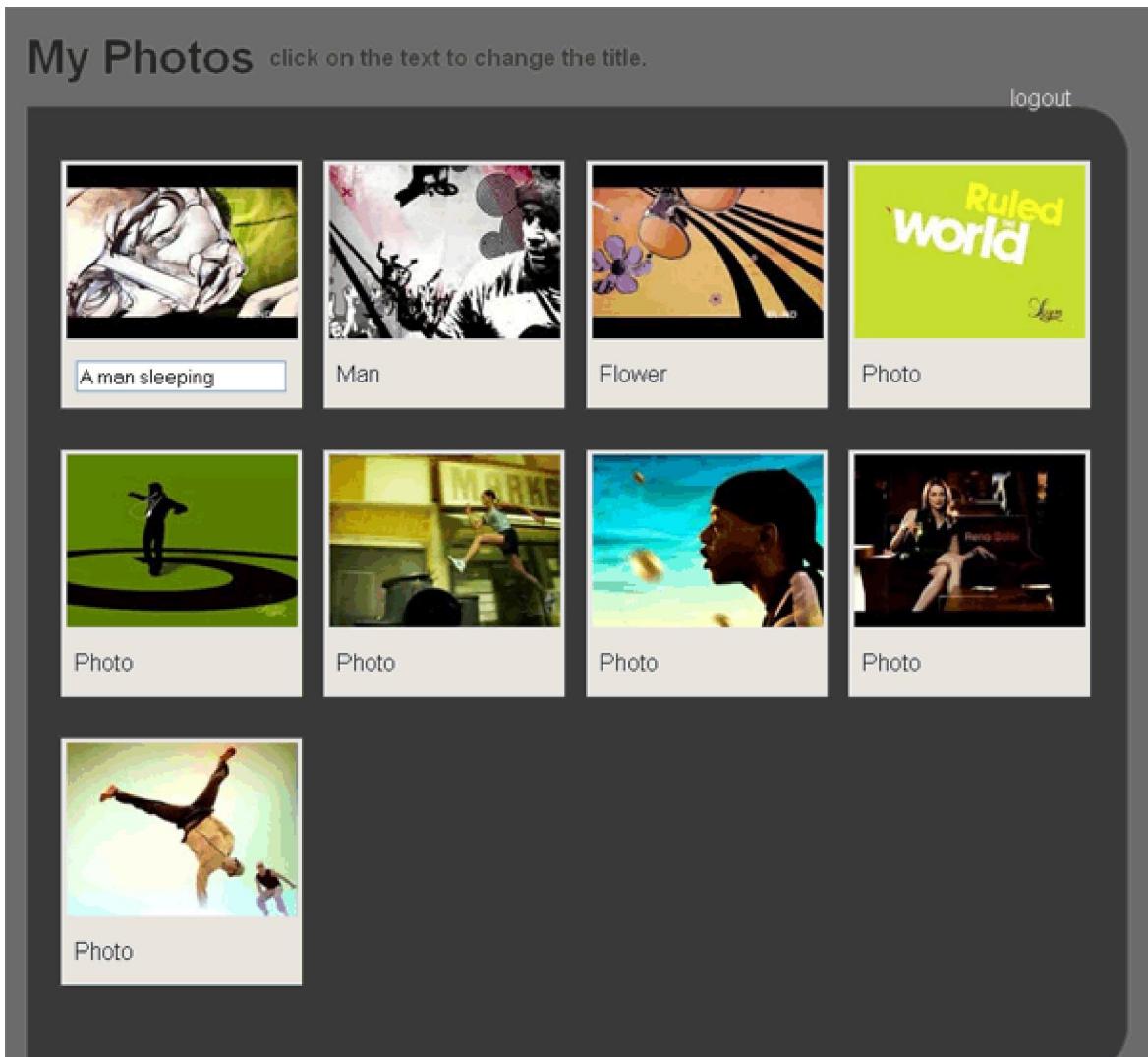
We'll store all of the information that we need in variables. For example, let's say that in one row, we have the following data.

- id = 4
- src = 10.jpg
- title = "My favorite photo"

By performing `$row->title`, we can grab that value and store it in our `$title` variable. So in this case, `$title = "My favorite photo"; $src = "10.jpg"; $id = 4;`.

All that we need to do now is add that information into our image and anchor tags. The last line will add an input field. This will eventually allow the user to update the title of the image asynchronously.

If you run "index.php" in your browser, you should see the following:



### AJAX Time

---

Now that we have the basic layout of our site, let's ajaxify it! We'll start by summarizing what we need our PHP to do. I find that this helps a great deal when working.



- When the user clicks on the title of a specific image, we'll then display an input field that allows the user to change the title.
- When he tabs away from that input field, we'll use AJAX to perform a SQL update statement.
- To give the user some feedback, we'll display a div that says something along the lines of "Success. The database has been updated."

Create a PHP file called "changePhotoTitle.php" and paste in the following code:

□

```
1. <?php
2.
3. require_once 'database.php';
4.
5. $title = mysql_real_escape_string($_POST['title']);
6. $id = mysql_real_escape_string($_POST['id']);
7.
8. $update_query = "UPDATE photos SET title = '$title' WHERE id='$id'";
9. $result = $mysqli->query($update_query) or die(mysqli_error($mysqli));
10.
11. if ($result) {
12.   echo "Success! <br />";
13.   echo "The title of this photo has been changed to: <strong>$title</strong>";
14. }
15. ?>
```

We've required the database file again. (Aren't we glad that we stored that in a separate file?) Next, we're creating two variables. \$title is equal to whatever the user enters into that input field. \$id is equal to its respective id field in the database.

To prevent some SQL injection, we'll wrap those post values with [mysql\\_real\\_escape\\_string\(\)](#).

#### *Updating the Database*

---

The update is rather straight forward. Update the photos table and change the title field to whatever the users enters in to that textbox; But only change the title field that has an id equal to \$id. (Refer to the screencast for a better understanding.)

If the update has been performed successfully, we'll echo a congratulations message.

### *Implementing the Javascript*

---

In order to perform the update without a postback, we'll use jQuery. Create a new Javascript file called "scripts.js" and paste in the following:

```
1. $(function() {  
2.     $('#photos input').change(function() {  
3.         $(this).slideUp().next('input').slideDown();  
4.     });  
5.  
6.     $('ul#photos input').change(function() {  
7.         var id = $(this).parent('li').find('img').attr('id');  
8.         var thisParam = $(this);  
9.         var title = $(this).val();  
10.  
11.        $.ajax({  
12.            type: 'POST',  
13.            url: 'changePhotoTitle.php',  
14.            data: 'title=' + title + '&id=' + id,  
15.            success: function(response) {  
16.                $('#response').fadeIn(1000).empty().append(response).prepend('<span id="x">X</span  
17.            >');  
18.            $('#span#x').click(function() {  
19.                $('#div#response').fadeOut('slow');  
20.            });  
21.        }  
22.    });  
23.});  
24.});  
25.});  
26.});  
27.});  
28.});  
29.});  
30.});  
31.});
```

- When the user clicks on the title of an image (the h4 tag), we'll run a function. We'll hide the h4 tag and then display the input field instead.
- We'll create two variables. "title" will store the value of what is entered into the textbox. "id" will store its respective id.

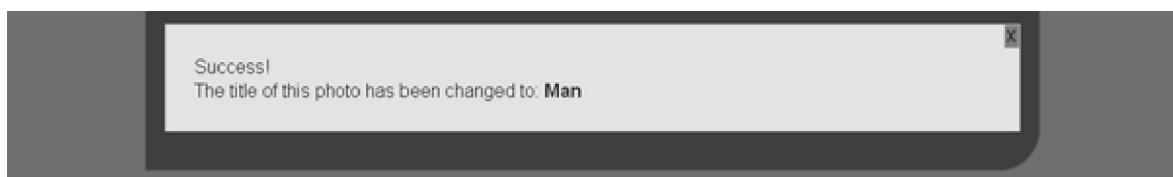
### **AJAX**

---

Using "\$.ajax", we'll call the update. \$.ajax will accept four parameters.

- Type = the type of update. In this case, it will be 'Post'
- Url = the url of the file that will perform our update. We've already created that file.
- Data = the information that we're going to pass to changePhotoTitle.php. In this case, we're passing the value of the textbox and the id.
- Success = this will be the function that runs if the update has been performed successfully. In this function, we'll remove the textbox because we no longer need it. We'll go back and display the h4 tag that now has the updated value.

To provide user feedback, we'll take the "response" div that we created at the beginning of this tutorial and fill it with our "Congratulations" echo. Those last two lines simply allow for a way for the user to remove that "Congratulations" div. If you need to, refer to the screencast for a full explanation.



### *Implementing the Login Form*

---

A screenshot of a light gray login form. At the top, it says "LOGIN" in bold capital letters. Below that are two input fields: one for "Username" and one for "Password", both with placeholder text. At the bottom is a blue "Login" button.

Now obviously, we don't want just anyone to be able to edit the titles of these images. So, we'll create a simple membership system.

Create a new page called "login.php" and add the following code:

```
1. <?php
2. if($_GET['logout'] == 1) {
3.     setcookie('authorized', 0, time()-3600);
4. }
5. ?>
6.
7.
8. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
9.
```

```

10. <html xmlns="http://www.w3.org/1999/xhtml">
11. <head>
12.   <title>Login to edit photos</title>
13.
14. <style type="text/css">
15. h2 {
16. margin-top: 0;
17. }
18.
19.
20. body {
21. text-align: center;
22. font-family: helvetica, arial;
23. }
24.
25. #loginForm {
26.
27. padding: 1em;
28. background: #e3e3e3;
29. width: 260px;
30. margin: 3em auto 0;
31. text-align: left;
32. }
33.
34. </style>
35. </head>
36.
37. <body>
38. <div id="loginForm">
39.   <form method="post" action="confirmLoginCredentials.php">
40.     <h2>LOGIN</h2>
41.     <p>Username: <input type="text" name="username" /></p>
42.     <p>Password: <input type="password" name="password" /></p>
43.     <p><input type="submit" name="submit" value="Login" /></p>
44.
45.   </form>
46. </div>
47. </body>
48. </html>

```

Within the body tag, we have a simple form that has two fields: one for the username, the other for the password. When the user clicks the "Login" button, the information from those textboxes will post to "confirmLoginCredentials.php". Go ahead, create that page right now and add the following.

```

1. <?php
2.
3. $username = mysql_real_escape_string($_POST['username']);

```

```

4. $password = mysql_real_escape_string($_POST['password']);
5.
6. require 'database.php';
7. $q = "SELECT first_name, last_name FROM users WHERE user_name = '$username' AND password = '$password'";
8.
9. $result = $mysqli->query($q) or die(mysql_error($mysqli));
10.
11.
12. if (!mysql_num_rows($result) == 1) {
13.   header("Location: login.php");
14. }
15. else {
16.   setcookie('authorized', 1, 0);
17.   header("Location: index.php");
18. }
19.
20. ?>

```

Per usual, we are storing the values from those textboxes in variables called \$username and \$password, respectively. We once again pull in our database.php file. Our query retrieves the row from the database where the user\_name and the password are equal to what was entered into those two textboxes. Only one row should be returned. If that doesn't happen, the user will be redirected back to "login.php". Otherwise, we'll send them to the main "index.php" page and set a cookie.

1. setcookie('authorized', 1, 0);

The name of our cookie will be "authorized"; the value will be equal to 1; and the expiration date will be 0. 0 essentially means that when the user closes his browser, that cookie will expire.

When the user is directed to the index.php page, we need to write a quick bit of code that ensures that there is the required cookie on the user's computer. Go back to index.php and add the following code to the very top of your document.

1. <?php
2. // Verifies if you have a cookie - credentials.
- 3.
4. if (!\$COOKIE['authorized'] == "1") {
5. header("Location: login.php");
6. }
7. ?>

If the user has a cookie named "authorized" on his computer, and that cookie has a value equal to 1, then display the page. **But if they DON'T, send them back to login.php.**

### *Logging Out*

---

The final step in this tutorial is to allow the user to log out. Find the tag on your index.php page that looks like this:

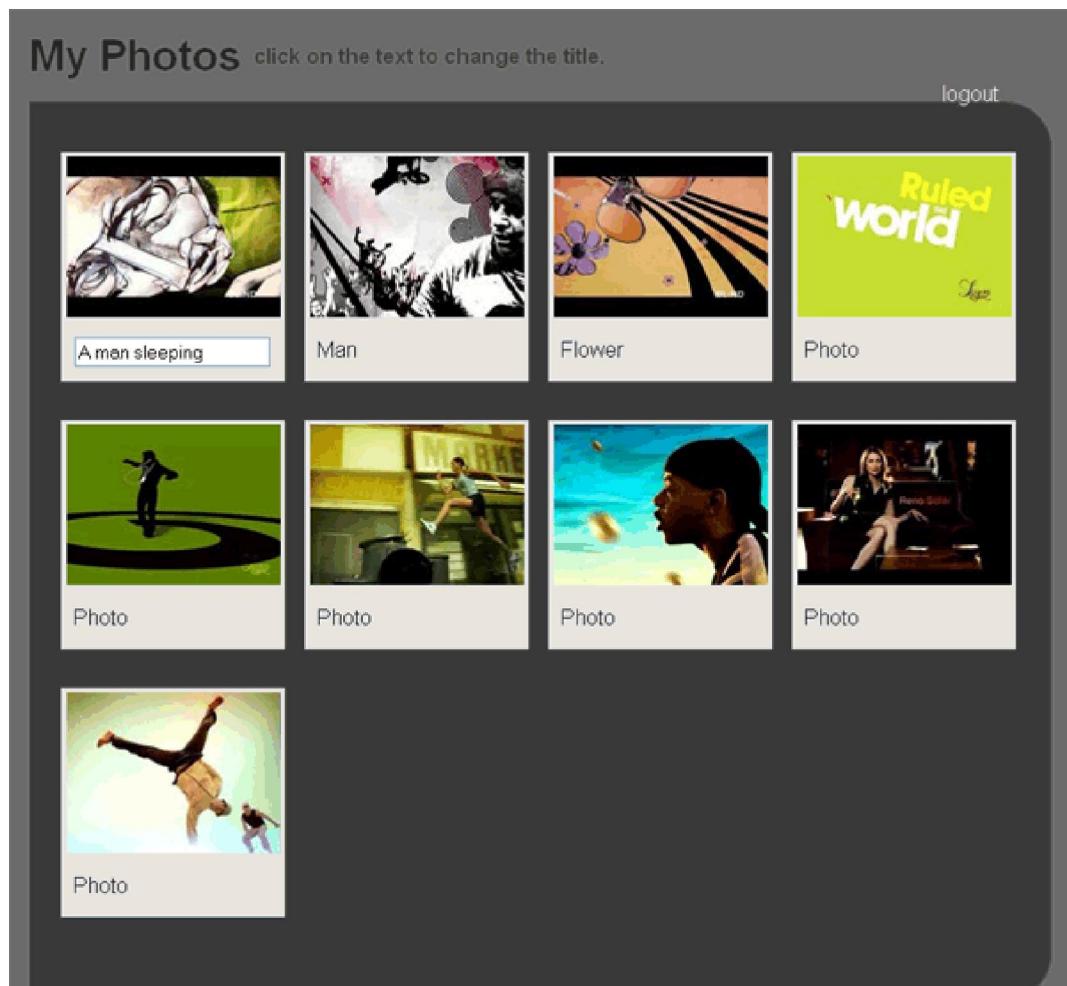
```
1. <a href="login.php?logout=1" id="logout">logout</a>
```

I'm adding a key-value pair to this url. That value will be passed to 'login.php'. Go to this page and add your last bit of code to the very top.

```
1. <?php  
2. if($_GET['logout'] == 1) {  
3.     setcookie('authorized', 0, time()-3600);  
4. }  
5. ?>
```

If, in the querystring, the key of 'logout' is equal to 1, set a cookie that has an expiration date of an hour ago. This will remove the cookie from the user's computer.

*Finally, you're done...for now!*



So we've created quite a bit. Unfortunately, to do everything I'd like would take pages and pages. Perhaps next week, Part 2 will be released. In Part 2, we'll implement the following:

- Write some code that allows the administrator to edit the photos. If they aren't logged in, they can view the photos, but aren't allowed to edit them.
- Implement Lightbox functionality
- Create insert and delete pages
- Secure the site more
- Add more validation
- Make the site prettier
- And plenty more...

Thanks so much for watching the screencast and reviewing the article. There's more to come!

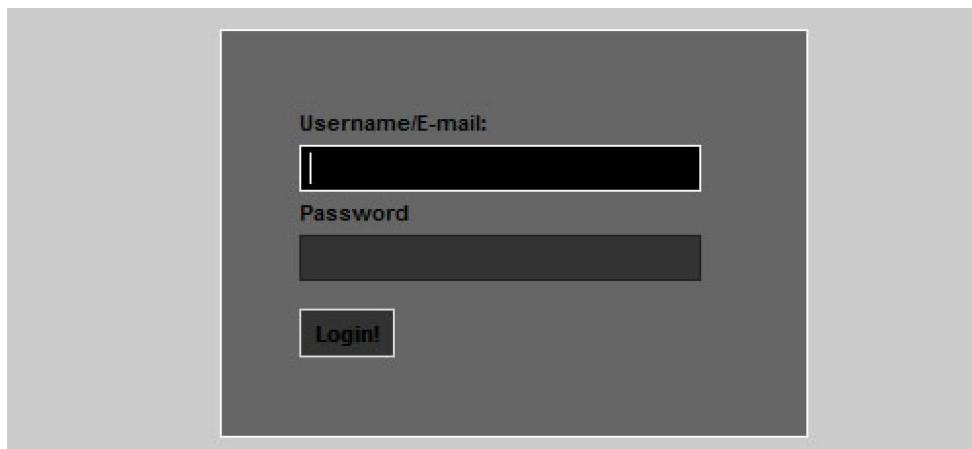
## 32. PHP ajax login form using Jquery

Because a lot of responses to the previous post php ajax tutorial create ajax based login form using jquery, and asking about some bug fixxing incl IE bugs, i decide to create a new PHP ajax login functionality using Jquery, and here it is...

### *Whats new?*

---

- Completely recoded
- Tested in IE6, FF3, Opera9 and i think should be worked in other modern browsers as well
- Easliy switch the data source between database and PHP array
- Login authentication can be using username or [email](#), or even both of 'em by changing the config
- Automattic redirection could be done within the javascript config
- Redirect target (on success) can be changed in config file
- Message and Text controled by array variable which could be changed in 1 [file](#)



### *dependency:*

---

- [Jquery 1.2.3](#), never tested with the prior version
- [PHP5](#), If you running prior version then you need to make some changes, probably but not limited at class constructor to make it compatible with your PHP version.

### *How to?*

---

- **Switch the data source** : Turn the value of USEDDB constant in login.config.php into true, and modify the db\_config array values as needed, to use array vars turn it false and modify the user\_config array values
- **change the authentication method** : change the LOGIN\_METHOD constant value, possible values:
  - user : match the username and password

- email : match the user email and password
- both : match the username or user email and password
- **switch automatic redirection** : turn the value of autoRedir var in login.js to true
- **change the redirect target** : change the value of SUCCESS\_URL constant in login.config.php, for example you need to automatically redirect to user profile, so change it into "index.php?profile=".\$\_SESSION['exp\_user']['username']
- **change error message and text** : change the array values accordingly at login.lang.php

#### *Notes:*

---

- Thanks to John resig and the jquery developer team, thanks guys you made my life easier
- Please do not link to the [download](#) file directly, link to this page instead
- This script is absolutely free, credits nor link is not required but will be great if this site got 'em

#### *Demo & Download*

---

- The [demo available here](#) use admin|admin@localhost|123456 as the test account
- [Download the source](#) again please do not link to the file directly, link to this page instead.

#### *SQL*

---

```
01.CREATE TABLE `tbl_user`  
02.(  
03.`userid` integer (11) NOT NULL AUTO_INCREMENT ,  
04.`username` varchar (50) NOT NULL,  
05.`useremail` varchar (100) NOT NULL,  
06.`userpassword` varchar (50) NOT NULL,  
07.PRIMARY KEY (`userid`)  
08.) TYPE=MyISAM CHARACTER SET utf8 COLLATE utf8_general_ci;  
09.  
10.BEGIN;  
11.INSERT INTO `tbl_user` (`userid`, `username`, `useremail`, `userpassword`)  
VALUES(1, 'admin', 'admin@chazzuka.com', 'e10adc3949ba59abbe56e057f20f883e');  
12.INSERT INTO `tbl_user` (`userid`, `username`, `useremail`, `userpassword`)  
VALUES(2, 'user', 'user@chazzuka.com', 'e10adc3949ba59abbe56e057f20f883e');  
13.COMMIT;
```

The Jquery file can be download at [jquery.com](http://jquery.com) If you found something doesnt work as expected just let me know.

### 33. HOWTO: PHP and jQuery upload progress bar

With the controllable jQuery Progress Bar, writing a form upload progress bar seems like a piece of cake now. Hypothetically, all we need is to create the bar, poll for the progress of the file upload, derive the new progress bar value (in percentage) and set it.

To do that you need to prepare the php script to do it. By default PHP can't report the progress of upload progress. However people smarter than me have already solved that problem. In 5 mins i've found 2 solutions: the [Alternative PHP Cache \(APC\)](#) method as well as the [UploadProgress](#) method. Both of them are PECL packages. Because i couldn't get APC to work on my server properly, i'll document the UploadProgress more in detail here...

**Step 1: Install the uploadprogress package.** Really simple just run the following command

```
pecl install uploadprogress
```

Once that is done, register the extension to your PHP with the following line in your php.ini

```
extension=uploadprogress.so
```

then restart your apache/httpd

**Step 2: Create the form and your progress bar**

```
<form id="uploadform" enctype="multipart/form-data" method="post">
<input id="progress_key" name="UPLOAD_IDENTIFIER" type="hidden" value="<?=$uuid ?>" />
<input id="ulfile" name="ulfile" type="file" />
<input type="submit" value="Upload" />
    <span id="uploadprogressbar" class="progressbar">0%</span>
</form>
```

this creates the form with a file field as well as a unique UPLOAD\_IDENTIFIER hidden field that allows our script to check the progress of the form submission.

**Step 3: Next the script itself to check the respond with the progress of the form submission.** Lets call this file *uploadprogress.php*

```
header ("Cache-Control: no-cache, must-revalidate");
header ("Expires: Mon, 26 Jul 1997 05:00:00 GMT");

if (@$_GET['id']) {
    echo json_encode(uploadprogress_get_info($_GET['id']));
    exit();
}
```

The header no-cache declarations circumvents IE's cache of the response. Basically this form does nothing but respond with a json encoded string of the uploadprogress\_get\_info function. The id argument is the same one we used in the form. Think of it as a form-upload-process-id. A typical response looks like this:

```
{"time_start":"1214384364","time_last":"1214384366","speed_average":"25889","speed_last":"40952","bytes_uploaded":"51778","bytes_total":"8125518","files_uploaded":"0","est_sec":"311"}
```

the response encodes a good deal of data about the form submission. most importantly for us: *bytes\_uploaded* and *bytes\_total*

### Step 3: use jQuery and a timer to keep polling the page and update the progress bar value

```
var progress_key = '';

// this sets up the progress bar
$(document).ready(function() {
    $("#uploadprogressbar").progressBar();
});

// fades in the progress bar and starts polling the upload progress after
// 1.5seconds
function beginUpload() {
    $("#uploadprogressbar").fadeIn();
    setTimeout("showUpload()", 1500);
}

// uses ajax to poll the uploadprogress.php page with the id
// deserializes the json string, and computes the percentage (integer)
// update the jquery progress bar
// sets a timer for the next poll in 750ms
function showUpload() {
    $.get("uploadprogress.php?id=" + progress_key, function(data) {
        if (!data)
            return;

        var response;
        eval ("response = " + data);

        if (!response)
            return;

        var percentage = Math.floor(100 *
parseInt(response['bytes_uploaded']) / parseInt(response['bytes_total'])));
        $("#uploadprogressbar").progressBar(percentage);

    });
    setTimeout("showUpload()", 750);
}
```

viola! read the comments if you dont understand the code. it is \_THAT\_ straightforward. Of course there can be many improvements such as stopping the script when the upload reaches

100% but that's probably not really needed since the whole page is refreshed. But this approach allows the flexibility of ajax submissions and what nots.

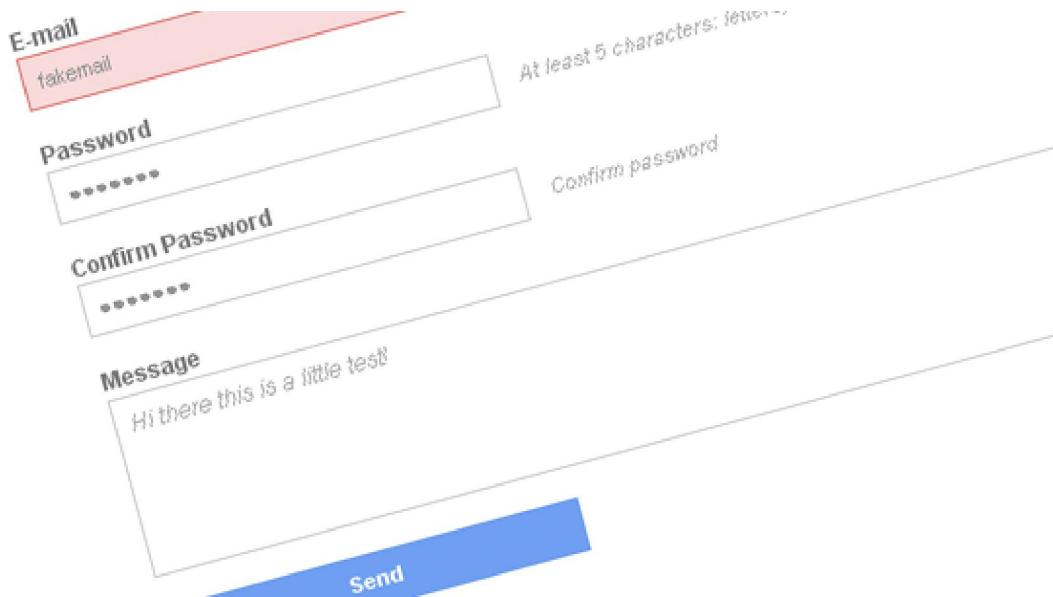
**Again, download the jQuery progressbar here: [jQuery progressbar](#) or view the demo [here](#)**

## 34. How to Validate Forms in both sides using PHP and jQuery

Hi guys and welcome to a new tutorial of yensdesign.com! Some of you ask us about the possibility of create a tutorial about how to validate forms in server side (using PHP). So i wrote a tutorial about it, adding some interesting features with [jQuery](#):

We are going to **learn how to validate your forms using PHP and jQuery in both sides: client side** (using javascript with [jQuery](#)) and **server side** (using PHP). It will be **interesting** to see **how to use regular expressions** to validate different kind of e-mails, passwords and more.

As always, here you have a preview of what we are going to do:



You can try the [living example](#) before continue reading the tutorial. Remember that **if you want to try the server side**, you need to **disable javascript in your web browser!**

Tested in: [Firefox](#), [Internet Explorer 6 & 7](#), [Opera](#), [Safari](#) & [Chrome](#).

Let's learn guys!

### INTRODUCTION

As you know, we are asking you [wich tutorials do you want to see](#) in yensdesign and as i told you, you can [send us more suggestions and feedback](#) in our forums.

On of these suggestions is about **how to validate forms in server side using PHP**, so in this tutorial we will make use of PHP and jQuery to validate all fields in both sides:

- **Client side:** Using javascript with [jQuery](#).
- **Server side:** Using PHP.

We will make use of regular expressions and if you don't know about them, you will notice very soon their power. By the way, you can check out [this source](#) to learn more about regular expressions in different languages.

At the end of the tutorial, **we will get a final real working example** that will **validate our fields with / without javascript enabled**, but **always making use of the server side validation**.

Let's dive into the tutorial!

#### **STEP 1: FIRST APPROACH TO THE XHTML LAYOUT**

In this case we will create a simple HTML layout showing our form. In a first approach the HTML will look like this:

```

1.   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2.   <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
3.     <head>
4.       <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
5.       <title>yensdesign.com - Validate Forms using PHP and jQuery</title>
6.       <link rel="stylesheet" href="css/general.css" type="text/css" media="screen" />
7.     </head>
8.     <body>
9.       <a id="logo" title="Go to yensdesign.com!" href="http://www.yensdesign.com"></a>
10.      <div id="container">
11.        <h1>Registration process</h1>
12.        <div id="error">
13.          <ul>
14.            <li><strong>Invalid Name:</strong> We want names with more than 3 letters!</li>
15.            <li><strong>Invalid E-mail:</strong> Stop cowboy! Type a valid e-mail please :P</li>
16.            <li><strong>Passwords are invalid:</strong> Passwords doesn't match or are invalid!</li>
17.            <li><strong>Invalid message:</strong> Type a message with at least 10 letters</li>
18.          </ul>
19.        </div>
20.        <div id="error" class="valid">
21.          <ul>
22.            <li><strong>Congratulations!</strong> All fields are OK ;)</li>
23.          </ul>
24.        </div>
25.

```

```

26.      <form method="post" id="customForm" action="">
27.          <div>
28.              <label for="name">Name</label>
29.              <input id="name" name="name" type="text" />
30.              <span id="nameInfo">What's your name?</span>
31.          </div>
32.          <div>
33.              <label for="email">E-mail</label>
34.              <input id="email" name="email" type="text" />
35.              <span id="emailInfo">Valid E-
    mail please, you will need it to log in!</span>
36.          </div>
37.          <div>
38.              <label for="pass1">Password</label>
39.              <input id="pass1" name="pass1" type="password" />
40.              <span id="pass1Info">At least 5 characters: letters, n
    umbers and '_</span>
41.          </div>
42.          <div>
43.              <label for="pass2">Confirm Password</label>
44.              <input id="pass2" name="pass2" type="password" />
45.              <span id="pass2Info">Confirm password</span>
46.          </div>
47.          <div>
48.              <label for="message">Message</label>
49.              <textarea id="message" name="message" cols="" rows="">
    </textarea>
50.          </div>
51.          <div>
52.              <input id="send" name="send" type="submit" value="Send
    " />
53.          </div>
54.      </form>
55.  </div>
56.  <script type="text/javascript" src="jquery.js"></script>
57.  <script type="text/javascript" src="validation.js"></script>
58. </body>
59. </html>

```

As you can see we have created all divisions that will appear (or not) in our tutorial. Some of these divisions won't be visible as default, but we will add some conditionals in PHP and CSS to get this. So these are the divisions:

- **#container:** contains all divisions
- **#error:** contains a list of possible errors after submitting the form
- **#error.valid:** contains the “congratulations phrase” if all it's OK
- **#customForm:** contains our form 😊

So now we have our xHTML layout ready let's add some style with CSS.

## STEP 2: ADDING STYLE WITH CSS

It will be fast guys, just take a look at the **general.css** code:

```
1. @CHARSET "UTF-8";
2. ***** GENERAL RESET *****/
3. html, body, div, span, applet, object, iframe, h1, h2, h3, h4, h5, h6,
p, blockquote, pre, a, abbr, acronym, address, big, cite, code, del, d
fn, em,
4. font, img, ins, kbd, q, s, samp, small, strike, strong, sub, sup, tt,
var, dl, dt, dd, ol, ul, li, fieldset, form, label, legend, table, capt
ion, tbody,
5. tfoot, thead, tr, th, td {
6. border:0pt none;
7. font-family:inherit;
8. font-size: 100%;
9. font-style:inherit;
10. font-weight:inherit;
11. margin:0pt;
12. padding:0pt;
13. vertical-align:baseline;
14. }
15. body{
16.     background: #fff;
17.     line-height:14px;
18.     font-size: 12px;
19.     font-family: Arial, Verdana, Helvetica, sans-serif;
20.     margin:0pt;
21.     cursor:default;
22.     overflow: hidden;
23. }
24. html,body{
25.     height:100%;
26.     text-align: center;
27. }
28. .clear{
29.     clear: both;
30.     height: 0;
31.     visibility: hidden;
32.     display: block;
33. }
34. a{
35.     text-decoration: none;
36. }
37. strong{
38.     font-weight: 700;
39. }
40. ***** GENERAL RESET *****/
41. h1{
42.     font-weight: 700;
43.     font-size: 18px;
44.     line-height: 1.2em;
45.     border-bottom: 1px dotted #6b9ef1;
46.     color: #5f95ef;
47.     margin-bottom: 1em;
48. }
49. ***** LOGO *****/
50. #logo{
```

```
51.     margin-top: 1em;
52.     display: block;
53. }
54. /***** /LOGO *****/
55. /***** CONTAINER *****/
56. #container{
57.     width: 600px;
58.     margin: 40px auto;
59.     text-align: left;
60. }
61. /***** /CONTAINER *****/
62. /***** FORM *****/
63. #customForm{
64.     padding: 0 10px 10px;
65. }
66. #customForm label{
67.     display: block;
68.     color: #797979;
69.     font-weight: 700;
70.     line-height: 1.4em;
71. }
72. #customForm input{
73.     width: 220px;
74.     padding: 6px;
75.     color: #949494;
76.     font-family: Arial, Verdana, Helvetica, sans-serif;
77.     font-size: 11px;
78.     border: 1px solid #cecece;
79. }
80. #customForm input.error{
81.     background: #f8dbdb;
82.     border-color: #e77776;
83. }
84. #customForm textarea{
85.     width: 550px;
86.     height: 80px;
87.     padding: 6px;
88.     color: #adaeae;
89.     font-family: Arial, Verdana, Helvetica, sans-serif;
90.     font-style: italic;
91.     font-size: 12px;
92.     border: 1px solid #cecece;
93. }
94. #customForm textarea.error{
95.     background: #f8dbdb;
96.     border-color: #e77776;
97. }
98. #customForm div{
99.     margin-bottom: 15px;
100. }
101. #customForm div span{
102.     margin-left: 10px;
103.     color: #b1b1b1;
104.     font-size: 11px;
105.     font-style: italic;
106. }
107. #customForm div span.error{
```

```

108.     color: #e46c6e;
109. }
110. #customForm #send{
111.     background: #6f9fff1;
112.     color: #fff;
113.     font-weight: 700;
114.     font-style: normal;
115.     border: 0;
116.     cursor: pointer;
117. }
118. #customForm #send:hover{
119.     background: #79a7f1;
120. }
121. #error{
122.     margin-bottom: 20px;
123.     border: 1px solid #efefef;
124. }
125. #error ul{
126.     list-style: square;
127.     padding: 5px;
128.     font-size: 11px;
129. }
130. #error ul li{
131.     list-style-position: inside;
132.     line-height: 1.6em;
133. }
134. #error ul li strong{
135.     color: #e46c6d;
136. }
137. #error.valid ul li strong{
138.     color: #93d72e;
139. }
140. /***** /FORM *****/

```

As always we make use of our cool CSS reset snippet and add some interesting parts to the specific tutorial.

Just notice that we have defined some **.error classes** that we will use in the [jQuery](#) part to make our tutorial a little more cool and friendly user 😊

Next part, the javascript validation!

### STEP 3: VALIDATION IN THE CLIENT SIDE WITH JQUERY

Yeah guys, because it could not be otherwise we will use [jQuery](#) to make the validation part in the client side. We will add / remove some .error classes to make our form a little more intuitive for the final user.

So first of all, remember that all the following code will be in the `$(document).ready()` of jQuery and all will be a part of the javascript file named **validation.js** (original, uh?).

So let's save **references** to some **DOM elements** that we will often:

```
1. //global vars
2. var form = $("#customForm");
3. var name = $("#name");
4. var nameInfo = $("#nameInfo");
5. var email = $("#email");
6. var emailInfo = $("#emailInfo");
7. var pass1 = $("#pass1");
8. var pass1Info = $("#pass1Info");
9. var pass2 = $("#pass2");
10. var pass2Info = $("#pass2Info");
11. var message = $("#message");
```

As you may notice, all references are form related.

Now let's define our validation functions that will help us in the validation process:

- **validateEmail()**: we only allow valid emails!
- **validateName()**: we only allow names with more than 3 letters
- **validatePass1()**: we only allow passwords with at least 5 characters
- **validatePass2()**: we only allow it if passwords are equal
- **validateMessage()**: we only messages with more than 10 letters

```
1. function validateName() {
2.     //if it's NOT valid
3.     if(name.val().length < 4){
4.         name.addClass("error");
5.         nameInfo.text("We want names with more than 3 letters!");
6.         nameInfo.addClass("error");
7.         return false;
8.     }
9.     //if it's valid
10.    else{
11.        name.removeClass("error");
12.        nameInfo.text("What's your name?");
13.        nameInfo.removeClass("error");
14.        return true;
15.    }
16. }
17. function validatePass1(){
18.     var a = $("#password1");
19.     var b = $("#password2");
20.
21.     //it's NOT valid
22.     if(pass1.val().length < 5){
23.         pass1.addClass("error");
24.         pass1Info.text("Ey! Remember: At least 5 characters: letters,
numbers and '_'");
25.         pass1Info.addClass("error");
26.         return false;
27.     }
}
```

```

28.      //it's valid
29.      else{
30.          pass1.removeClass("error");
31.          pass1Info.text("At least 5 characters: letters, numbers and '_
  ");
32.          pass1Info.removeClass("error");
33.          validatePass2();
34.          return true;
35.      }
36.  }
37.  function validatePass2(){
38.      var a = $("#password1");
39.      var b = $("#password2");
40.      //are NOT valid
41.      if( pass1.val() != pass2.val() ){
42.          pass2.addClass("error");
43.          pass2Info.text("Passwords doesn't match!");
44.          pass2Info.addClass("error");
45.          return false;
46.      }
47.      //are valid
48.      else{
49.          pass2.removeClass("error");
50.          pass2Info.text("Confirm password");
51.          pass2Info.removeClass("error");
52.          return true;
53.      }
54.  }
55.  function validateMessage(){
56.      //it's NOT valid
57.      if(message.val().length < 10){
58.          message.addClass("error");
59.          return false;
60.      }
61.      //it's valid
62.      else{
63.          message.removeClass("error");
64.          return true;
65.      }
66.  }

```

As you can see, we are **adding and removing .error classes** and **changing text values** if needed to **show the user what fields are wrong**.

Now we have defined all validation functions, we only need to **manage events**. Here you have what we are going to do:

- Validate the **name field** in: **blur** and **keyup** events.
- Validate the **email field** in: **blur** event.
- Validate the **password fields** in: **blur** and **keyup** events.
- Validate the **message field** in: **blur**, and **keyup** event.
- Validate **all fields** in: **submit form event**.

Therefore the code would be:

```
1. //On blur
2. name.blur(validateName);
3. email.blur(validateEmail);
4. pass1.blur(validatePass1);
5. pass2.blur(validatePass2);
6. //On key press
7. name.keyup(validateName);
8. pass1.keyup(validatePass1);
9. pass2.keyup(validatePass2);
10. message.keyup(validateMessage);
11. //On Submitting
12. form.submit(function() {
13.     if(validateName() & validateEmail() & validatePass1() & validatePa
ss2() & validateMessage())
14.         return true
15.     else
16.         return false;
17. });


```

As you can see, it's very easy to control whatever we want by using jQuery and some functions. So now that we have done the client site, **let's move to the server side!**

#### STEP 4: VALIDATION IN THE SERVER SIDE WITH PHP

**Now that we have the client side done, let's use the power of PHP to complete our tutorial.**

Some of you may **think that we don't need the server side validation** but it's a big error thinking about it. Remember that **we can't trust** in the client side. In example: **users can disable javascript** in their browsers and **send us undesired values**.

So first of all we must **create** a new .php file named **validation.php** that will **contains** our **validation functions** written in PHP:

```
1. <?php
2.     function validateName($name) {
3.         //if it's NOT valid
4.         if(strlen($name) < 4)
5.             return false;
6.         //if it's valid
7.         else
8.             return true;
9.     }
10.    function validateEmail($email){
11.        return ereg("^[a-zA-Z0-9]+[a-zA-Z0-9_-]+@[a-zA-Z0-9.-]+[a-zA-Z0-9_-]+.[a-zA-Z]{2,4}$", $email);
12.    }
13.    function validatePasswords($pass1, $pass2) {
14.        //if DOESN'T MATCH
```

```

15.         if(strpos($pass1, ' ') !== false)
16.             return false;
17.         //if are valid
18.         return $pass1 == $pass2 && strlen($pass1) > 5;
19.     }
20.     function validateMessage($message) {
21.         //if it's NOT valid
22.         if(strlen($message) < 10)
23.             return false;
24.         //if it's valid
25.         else
26.             return true;
27.     }
28. ?>

```

As you can see we have defined 4 functions:

- **validateName()**: checks if name has at least 4 letters
- **validateEmail()**: checks if email is valid
- **validatePasswords()**: checks if passwords are equal and have at least 5 letters
- **validateMessage()**: checks if message has at least 10 letters

It's almost done guys, we only need to add some conditionals to our **index.php**. If you remember, we said that we are going to add some conditionals in the PHP, so it's time to do that:

- We will **display #error division** only if there are at least one error.
- We will **display #error.valid division** only if all fields are OK.

So here you have the final **index.php** layout:

```

1.  <?php
2.      require_once("validation.php");
3.  ?>
4.
5.  <!DOCTYPE html PUBLIC "-//
//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD
/xhtml1-transitional.dtd">
6.  <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
7.  <head>
8.      <meta http-equiv="Content-Type" content="text/html; charset=UTF-
8" />
9.      <title>yensdesign.com -
Validate Forms using PHP and jQuery</title>
10.     <link rel="stylesheet" href="css/general.css" type="text/css" medi
a="screen" />
11. </head>
12. <body>
13.     <a id="logo" title="Go to yensdesign.com!" href="http://www.yensde
sign.com"></a>
14.     <div id="container">
15.         <h1>Registration process</h1>

```

```

16.
17.      <?if( isset($_POST['send']) &amp;amp;amp;amp;amp;amp;amp;amp;amp;amp; &amp;
18.          amp;amp;amp;amp;amp;amp; (!validateName($_POST['name']) || !validateEmail($_POST['email']) || !validatePasswords($_POST['pass1'], $_POST['pass2']) || !validateMessage($_POST['message'])) ) :?>
19.          <div id="error">
20.              <ul>
21.                  <?if(!validateName($_POST['name'])) :?>
22.                      <li><strong>Invalid Name:</strong> We want
23.                          names with more than 3 letters!</li>
24.                  <?endif?>
25.                  <?if(!validateEmail($_POST['email'])) :?>
26.                      <li><strong>Invalid E-
27.                          mail:</strong> Stop cowboy! Type a valid e-mail please :P</li>
28.                  <?endif?>
29.                  <?if(!validatePasswords($_POST['pass1'], $_POST['pass2'])) :?>
30.                      <li><strong>Passwords are invalid:</strong>
31.                          > Passwords doesn't match or are invalid!</li>
32.                      <?endif?>
33.                      <?if(!validateMessage($_POST['message'])) :?>
34.                          <li><strong>Invalid message:</strong> Type
35.                              a message with at least with 10 letters</li>
36.                          <?endif?>
37.                      </ul>
38.                  </div>
39.          <?elseif(isset($_POST['send'])) :?>
40.              <div id="error" class="valid">
41.                  <ul>
42.                      <li><strong>Congratulations!</strong> All fiel
43.                          ds are OK ;)</li>
44.                  </ul>
45.              </div>
46.          <?endif?>
47.
48.          <form method="post" id="customForm" action="">
49.              <div>
50.                  <label for="name">Name</label>
51.                  <input id="name" name="name" type="text" />
52.                  <span id="nameInfo">What's your name?</span>
53.              </div>
54.              <div>
55.                  <label for="email">E-mail</label>
56.                  <input id="email" name="email" type="text" />
57.                  <span id="emailInfo">Valid E-
58.                      mail please, you will need it to log in!</span>
59.              </div>
60.              <div>
61.                  <label for="pass1">Password</label>
62.                  <input id="pass1" name="pass1" type="password" />
63.                  <span id="pass1Info">At least 5 characters: letters, n
64.                      umbers and '_ '</span>
65.              </div>
66.              <div>
67.                  <label for="pass2">Confirm Password</label>
68.                  <input id="pass2" name="pass2" type="password" />
69.                  <span id="pass2Info">Confirm password</span>

```

```

62.          </div>
63.          <div>
64.              <label for="message">Message</label>
65.              <textarea id="message" name="message" cols="" rows="">
66.          </div>
67.          <div>
68.              <input id="send" name="send" type="submit" value="Send
" />
69.          </div>
70.      </form>
71.  </div>
72.  <script type="text/javascript" src="jquery.js"></script>
73.  <script type="text/javascript" src="validation.js"></script>
74. </body>
75. </html>

```

As you can see we have **included the validation.php file before call our validation functions** and added some **calls** to our validation functions to check fields when **form is submited**.

We have used **short notation PHP** in conditionals to **keep the HTML cleaner and easier to read**.

We are **showing errors / “congratulations phrase” as needed** and now **you only need to know what are you doing to do with this validated values** 😊

#### **STEP 5: TESTING OUR VALIDATED FORM!**

That's all guys, I hope you find it useful and use this tutorial to improve a little more your websites.

Remember that **if you want to try the server side, you need to disable the javascript in your web browser**. In Firefox it's in Tools > Options > Content uncheck the “Enable Javascript” checkbox.

You can try the [tutorial online here](#) to see how It is working and [download here all sources](#).

Remember that you can solve your doubts in [our forums](#) and [follow us on Twitter](#) to know more about what are we doing in any moment!

See you on next tutorial and remember that we want you suggest tutorials to write!

## 35. Auto-Complete Field with jQuery, JSON & PHP

In the last few days I've started using the jQuery JavaScript library. To experiment with this great piece of software I've decided to implement an AJAX auto-complete feature. jQuery makes remote scripting a piece of cake and that led me to spend more time coding additional functionalities for the auto-complete field. In this post I'll explain how to use my auto-complete field and in a following post I'll explain all the code. So let's start!

Before continuing let's see how the auto-complete field looks like. Start typing some color name in the input field bellow to see how it works:



Pretty hugh? Besides the auto-complete code we need the jQuery library along with its Dimensions plug-in (all files are linked at the end of this post). Let's include the files into our page:

```
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" src="dimensions.js"></script>
<script type="text/javascript" src="autocomplete.js"></script>
```

Now we need to call the function that will bring life to our auto-complete field - *setAutoComplete*.

```
<script type="text/javascript">
$(function() {
    setAutoComplete("searchField", "results", "autocomplete.php?part=");
});
</script>
```

The call to *setAutoComplete* is inside a jQuery special code that is only executed when the DOM is ready, or in other words, when all the code is already loaded. The *setAutoComplete* function takes 3 parameters:

- the id of the input field
- the id of the div that will hold the returned data
- the URL of the remote script that will process the request

Be aware that the URL should reflect your remote script and that it will be combined with the text typed in the input field.

Now we include our stylesheet to define the look of the elements. We need to define the styles of the div that will contain the results, they include two classes for the selected and unselected items. Take a look at the CSS file linked at the end of this post. There is also a style for the input field.

```
<link rel="stylesheet" href="autocomplete.css" type="text/css">
```

To finish the client side part now we need to define the code of the input field as follows:

```
<label for="searchField">Colors: </label>
<input type="text" id="searchField" name="searchField">
```

We are almost there! Now that the client side is finished it's time for the server script. Here is an example of a script that try to match colors. I'm using PHP but you can use whatever language you prefer for the job, of course. You just need to return the results as an array encoded as JSON.

```
// define the colors array
$colors = array('black', 'blue', 'brown', 'green', 'grey',
                'gold', 'navy', 'orange', 'pink', 'silver',
                'violet', 'yellow', 'red');

// check the parameter
if(isset($_GET['part']) and $_GET['part'] != '')
{
    // initialize the results array
    $results = array();

    // search colors
    foreach($colors as $color)
    {
        // if it starts with 'part' add to results
        if( strpos($color, $_GET['part']) === 0 ){
            $results[] = $color;
        }
    }

    // return the array as json
    echo json_encode($results);

    // or return using Zend_Json class
    // require_once('Zend/Json/Encoder.php');
    // echo Zend_Json_Encoder::encode($results);
}
```

Done! As said before, we return the data as an array encoded as JSON. If you are running PHP  $\geq 5.2.0$  or the json extension you simply use `json_encode` for the job. Another option is to use `Zend_Json_Encoder` from the Zend Framework!

You should have a working auto-complete field by now! Note that we don't need to explicit create the result div because it'll be created automatically. You should also note that the current script can be improved a lot and is limited to just one auto-complete field per page!

Keep tuned, in the next post I'll explain how the JavaScript works!  
Thank you!

## 36. Create a shoutbox using PHP and AJAX (with jQuery)

We will learn how to create a **dynamic ajax based shoutbox** with **jQuery from scratch**. It will be very interesting to know **how to use the ajax function of jQuery** and **how it can be used to insert and recover data from a MySQL database via PHP** in a way **asynchronous**.

**HERE YOU HAVE A PREVIEW IMAGE:**



You can try the [living example](#) before continue reading the tutorial.

**Tested in:** [Firefox](#), [Internet Explorer 6 \(bad display of PNG icons, IE bug\) & 7](#), [Opera](#), [Safari](#) & [Chrome](#).

Come on guys, let's dive into the tutorial!

### **WHAT WE WILL DO**

For those who don't know what is a shoutbox here you have a wikipedia dixit:

A **shoutbox**, **saybox**, **tagboard**, or **chatterbox** is a chat-like feature of some websites that allows people to quickly leave messages on the website, generally without any form of user registration.

Now that we know what we want to do, we will remember what we will use to achieve our purpose:

- xHTML & CSS
- PHP (We need a PHP server!)
- MySQL (We will need a MySQL database too!)
- Javascript & AJAX (via jQuery, of course!)

Don't worry if you don't know a lot about MySQL, We only need it to create a small table and to inser & recover data in a simply way via PHP.

Let's face the step 1 guys!

### **STEP 1: THE LAYOUT**

First of all we will create the layout (as always, you know), it will be similar to the [tabbed menu tutorial](#) and it will have 2 main divisions:

- Form (fields with the user and message to send)
- Container division (contains the messages)
- Loading division (as a part of the Container division, displays the loading animated gif... while messages are loading)

So now we have have seen the main divisions, take a look at the xHTML code:

```
1.  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2.  <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
3.  <head>
4.      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
5.      <title>yensdesign.com - Create a shoutbox using PHP and AJAX with jQuery</title>
6.      <link rel="stylesheet" href="css/general.css" type="text/css" media="screen" />
7.  </head>
8.  <body>
9.      <a id="logo" title="Go to yensdesign.com!" href="http://www.yensdesign.com"></a>
10.     <form method="post" id="form">
11.         <table>
12.             <tr>
13.                 <td><label>User</label></td>
14.                 <td><input class="text user" id="nick" type="text" MAXLENGTH="25" /></td>
15.             </tr>
16.             <tr>
17.                 <td><label>Message</label></td>
18.                 <td><input class="text" id="message" type="text" MAXLENGTH="255" /></td>
19.             </tr>
20.             <tr>
21.                 <td></td>
22.                 <td><input id="send" type="submit" value="Shout it!" /></td>
23.             </tr>
24.         </table>
25.     </form>
26.     <div id="container">
27.         <ul class="menu">
```

```

28.          <li>Shoutbox</li>
29.      </ul>
30.      <span class="clear"></span>
31.      <div class="content">
32.          <h1>Latest Messages</h1>
33.          <div id="loading"></div>
34.          <ul>
35.              <ul>
36.                  </div>
37.          </div>
38.          <script type="text/javascript" src="jquery.js"></script>
39.          <script type="text/javascript" src="shoutbox.js"></script>
40.      </body>
41.  </html>

```

We are **limiting** the **maxlength** of both fields: **nick(25)** & **message(255)** because our **shoutbox table** will have these length limitations.

Nothing more to explain over here, let's go to the CSS part.

#### STEP 2: ADDING SOME STYLE WITH CSS

There is nothing special in this case for the css part, just remember you that we are reusing the CSS trick for the tab that we have used in the [tabbed menu tutorial](#).

Here you have the entire CSS code:

```

1.  @CHARSET "UTF-8";
2.  /****** GENERAL RESET *****/
3.  html, body, div, span, applet, object, iframe, h1, h2, h3, h4, h5, h6,
p, blockquote, pre, a, abbr, acronym, address, big, cite, code, del, d
fn, em,
4.  font, img, ins, kbd, q, s, samp, small, strike, strong, sub, sup, tt,
var, dl, dt, dd, ol, ul, li, fieldset, form, label, legend, table, capt
ion, tbody,
5.  tfoot, thead, tr, th, td {
6.  border:0pt none;
7.  font-family:inherit;
8.  font-size: 100%;
9.  font-style:inherit;
10. font-weight:inherit;
11. margin:0pt;
12. padding:0pt;
13. vertical-align:baseline;
14. }
15. body{
16.     background: #fff;
17.     line-height:14px;
18.     font-size: 12px;
19.     font-family: Arial, Verdana, Helvetica, sans-serif;

```

```
20.      margin:0pt;
21.      cursor:default;
22.      overflow: hidden;
23. }
24. html,body{
25.     height:100%;
26.     text-align: center;
27. }
28. .clear{
29.     clear: both;
30.     height: 0;
31.     visibility: hidden;
32.     display: block;
33. }
34. a{
35.     text-decoration: none;
36. }
37. strong{
38.     font-weight: 700;
39. }
40. /***** GENERAL RESET *****/
41. /***** LOGO *****/
42. #logo{
43.     margin-top: 1em;
44.     display: block;
45. }
46. /***** /LOGO *****/
47. /***** FORM *****/
48. #form{
49.     margin: 5em auto 3em;
50.     width: 300px;
51. }
52. #form table td{
53.     text-align: left;
54.     font-size: 11px;
55. }
56. #form label{
57.     font-weight: 700;
58. }
59. #form input{
60.     border: 1px solid #d0ccc9;
61.     background: #fff;
62.     color: #5f95ef;
63.     font-size: 11px;
64.     font-weight: 700;
65. }
66. #form input.text{
67.     font-weight: normal;
68.     color: #565656;
69.     border: 1px solid #9c9c9c;
70.     width: 250px;
71.     padding: 2px;
72.     margin-bottom: 5px;
73.     text-align: left;
74. }
75. #form input.text.user{
76.     width: 100px;
```

```

77. }
78. /***** /FORM *****/
79. /***** MENU *****/
80. #container{
81.     margin: 1em auto;
82.     width: 400px;
83. }
84. #container ul{
85.     list-style: none;
86.     list-style-position: outside;
87. }
88. #container ul.menu li{
89.     float: left;
90.     margin-right: 5px;
91.     margin-bottom: -1px;
92. }
93. #container ul.menu li{
94.     font-weight: 700;
95.     display: block;
96.     padding: 5px 10px 5px 10px;
97.     background: #fff;
98.     border: 1px solid #d0ccc9;
99.     border-width: 1px 1px 0 1px;
100.    position: relative;
101.    color: #5f95ef;
102.    cursor: pointer;
103. }
104. /***** /MENU *****/
105. /***** CONTENT *****/
106. .content{
107.     margin: 0pt auto;
108.     background: #efefef;
109.     background: #fff;
110.     border: 1px solid #d0ccc9;
111.     text-align: left;
112.     padding: 10px;
113.     padding-bottom: 20px;
114.     font-size: 11px;
115. }
116. .content h1{
117.     line-height: 1em;
118.     vertical-align: middle;
119.     height: 48px;
120.     padding: 10px 10px 10px 52px;
121.     font-size: 32px;
122.     background: transparent url(images/bubble.jpg) no-
repeat scroll left top;
123. }
124. .date{
125.     font-weight: normal;
126.     font-size: 9px;
127.     color: #aeaeae;
128. }
129. /***** /CONTENT *****/
130. /***** LOADING *****/
131. #loading{
132.     text-align: center;

```

```
133. }
134. /****** /LOADING *****/
```

As you may noticed, we have an empty **ul element** with no **li elements**, that's because **we will load his content via AJAX** 😊

Let's create the MySQL table.

#### STEP 3: CREATING SHOUTBOX TABLE IN MYSQL

As I told you, **we will make use of a MySQL database to store our messages** and then **we will retrieve data via PHP**. So we only need to create a simple **table** called **shoutbox** with these columns:

- id (int primary key and with auto increment)
- date (timestamp default CURRENT\_TIMESTAMP)
- user (varchar 25)
- message (varchar 255)

So here you have the MySQL code that you must execute in your database (via phpmyadmin for example):

```
1. CREATE TABLE `shoutbox`(
2.     `id` int(5) NOT NULL auto_increment,
3.     `date` timestamp NOT NULL default CURRENT_TIMESTAMP,
4.     `user` varchar(25) NOT NULL default 'anonymous',
5.     `message` varchar(255) NOT NULL default '',
6.     PRIMARY KEY (`id`)
7. );
```

So now that we have our layout ready and the MySQL table created let's continue with the PHP part.

#### STEP 4: USING PHP TO INSERT & RECOVER DATA FROM MYSQL DATABASE

**Last two steps** (step 4 & step 5) are very **important** so guys **pay attention!** With **jQuery.ajax()** function from [jQuery](#) we can load a remote page using an HTTP request.

We will make use of the **jQuery.ajax()** function to make a **POST request** to a **PHP script** named **shoutbox.php** that **will manage the insert / retrieve data action** with a **switch sentence**. We will **receive a XMLHttpRequest object** from **\$.ajax()** function that will have a property called **responseText**. This property will give us the required information to display the shoutbox's messages (in case of select action) or the confirmation (in case of insert action).

So before add the AJAX code we need to create our **shoutbox.php** that will manage all the requests from our future javascript file **shoutbox.js**. We will define some constants to the MySQL connection:

```
1.  ****
2.      CONSTANTS
3.  ****
4.  define("HOST", "YOUR HOST");
5.  define("USER", "YOUR USER");
6.  define("PASSWORD", "YOUR USER PASSWORD");
7.  define("DB", "YOUR DATABASE");
```

As you may noticed the **second parameter of each define()** function it's wrong / **empty**, you **must change them for values** that apply in your case.

Now we have defined our constants to the connection with MySQL we will create 3 functions:

- **connect()**: It will open a **connection** to a specific host selecting a specific **database** too.
- **getContent()**: It will **retrieve** the last messages from our shoutbox table in plain text.
- **insertMessage()**: It will **insert** the new message in our shoutbox table.

So here you have the three functions:

```
1.  ****
2.      FUNCTIONS
3.  ****
4.  function connect($db, $user, $password) {
5.      $link = @mysql_connect($db, $user, $password);
6.      if (!$link)
7.          die("Could not connect: ".mysql_error());
8.      else{
9.          $db = mysql_select_db(DB);
10.         if(!$db)
11.             die("Could not select database: ".mysql_error());
12.         else return $link;
13.     }
14. }
15. function getContent($link, $num) {
16.     $res = @mysql_query("SELECT date, user, message FROM shoutbox ORDER
17. BY date DESC LIMIT ".$num, $link);
18.     if(!$res)
19.         die("Error: ".mysql_error());
20.     else
21.         return $res;
22. }
23. function insertMessage($user, $message) {
24.     $query = sprintf("INSERT INTO shoutbox(user, message) VALUES ('%s',
25. '%s')", mysql_real_escape_string(strip_tags($user)), mysql_real_escape_string(strip_tags($message)));
26.     $res = @mysql_query($query);
27.     if(!$res)
```

```

26.         die("Error: ".mysql_error());
27.     else
28.         return $res;
29. }

```

It's almost finished guys. As we said, we will **manage** via **switch sentence** the **POST requests from AJAX (from the future shoutbox.js file)** so we will **make use of the three previous functions to retrieve and insert data**:

```

1.  ****
2.      MANAGE REQUESTS
3.  ****
4.  if(!$_POST['action']){
5.      //We are redirecting people to our shoutbox page if they try to enter in our shoutbox.php
6.      header ("Location: index.html");
7.  }
8. else{
9.     $link = connect(HOST, USER, PASSWORD);
10.    switch($_POST['action']){
11.        case "update":
12.            $res = getContent($link, 20);
13.            while($row = mysql_fetch_array($res)){
14.                $result .= "
15.
16.                <li><strong>".$row['user']."'</strong><img src=\"\" alt=\"\"-
17.                \\" />".$row['message']."' <span class=\"date\">".$row['date']."'</span
18.                ></li>
19.            ";
20.            echo $result;
21.            break;
22.        case "insert":
23.            echo insertMessage($_POST['nick'], $_POST['message']);
24.            break;
25.    }
26.    mysql_close($link);
27. }

```

And that's all in the PHP part guys.

Just comment that we are **avoiding the direct access to shoutbox.php by checking the post variable action that we send in each request to shoutbox.php from javascript code (from shoutbox.js remember)**.

Let's go to the final step... the AJAX part!

#### STEP 5: ADDING AJAX AND SOME MAGIC WITH JQUERY

So now we have done the layout with the CSS, the shoutbox table in a MySQL database and the PHP script to insert / retrieve data.

We only need to know how to make POST requests from our shoutbox.js javascript file to our soutbox.php PHP script file.

All the following code will be in a javascript file named **shoutbox.js** and in the **\$(document).ready** of jQuery. First of all we will define some variables to save some references to jQuery selectors:

```
1. //global vars
2. var inputUser = $("#nick");
3. var inputMessage = $("#message");
4. var loading = $("#loading");
5. var messageList = $(".content > ul");
```

We will need a function to check if all fields are filled before send the data to our database, it will be called **checkForm()**:

```
1. //check if all fields are filled
2. function checkForm(){
3.     if(inputUser.attr("value") &amp; &amp;amp; &amp;amp; &amp; inputMessage
    .attr("value"))
4.         return true;
5.     else
6.         return false;
7. }
```

As we said in the Step 1, we will load the messages of our shoutbox table via AJAX from the first instance, so let's create a function to retrieve this data:

```
1. function updateShoutbox(){
2.     //just for the fade effect
3.     messageList.hide();
4.     loading.fadeIn();
5.     //send the post to shoutbox.php
6.     $.ajax({
7.         type: "POST", url: "shoutbox.php", data: "action=update",
8.         complete: function(data){
9.             loading.fadeOut();
10.            messageList.html(data.responseText);
11.            messageList.fadeIn(2000);
12.        }
13.    });
14. }
```

What's happening here? We are **showing the loading animated gif while we are retrieving data from shoutbox table** via POST request to our shoutbox.php.

With the **data parameter of \$.ajax() function** we can **send post variables (action = update)**, in this case **we are sending \$\_POST['action'] = "update"**, so **shoutbox.php** can **manage the specific action** in the **switch sentence** that we have made before.

Finally **when the request is completed**, we **hide the loading animated gif, load the new data** in our **xHTML layout** and **show it**.

So adding a call to this function, **updateShoutbox()** in the ready event of jQuery we can load the data in a first instance:

```
1. //Load for the first time the shoutbox data
2. updateShoutbox();
```

Finally we need to manage the submit event to insert user's messages in our shoutbox table and then refresh the content in the shoutbox (with the updateShoutbox() function, you know):

```
1. //on submit event
2. $("#form").submit(function(){
3.     if(checkForm()){
4.         var nick = inputUser.attr("value");
5.         var message = inputMessage.attr("value");
6.         //we deactivate submit button while sending
7.         $("#send").attr({ disabled:true, value:"Sending..." });
8.         $("#send").blur();
9.         //send the post to shoutbox.php
10.        $.ajax({
11.            type: "POST", url: "shoutbox.php", data: "action=insert&am
    p;nick=" + nick + "&message=" + message,
12.            complete: function(data){
13.                messageList.html(data.responseText);
14.                updateShoutbox();
15.                //reactivate the send button
16.                $("#send").attr({ disabled:false, value:"Shout it!" })
17.            }
18.        });
19.    }
20.    else alert("Please fill all fields!");
21.    //we prevent the refresh of the page after submitting the form
22.    return false;
23.});
```

As you can see we are making use of some previously defined functions like **checkForm()** to check if all fields are filled and **updateShoutbox()** to refresh the shoutbox content. In addition, we **have disabled the submit button** of our form **till the process isn't completed**.

In this case **we are sending more information** in the **data parameter** of the **`\$.ajax()`** function: **nick** and **message**, it will allow us to **save in our shoutbox table via php** the values of **`\$_POST['nick']`** and **`\$_POST['message']`**.

It's important to know that if we have more than one variable in **data parameter**, we need to separate them by using the “**&**” symbol, like in **GET parameters**.

All done guys!

#### **STEP 6: TESTING OUR AMAZING SHOUTBOX!**

And that's all guys, as you can see it's easy to achieve our goals if we want to learn 😊

That's all guys, I hope you enjoy this tutorial!

You can try the [tutorial online here](#) to see how It is working and [download here all sources](#).

One more thing guys! Remember that you can solve your doubts in [our forums](#) and [follow os on Twitter](#) to know more about what are we doing in any moment!

See you on next tutorial and remember that we want you [suggest tutorials](#) to write!

## 37. Tutorial - CakePHP Ajax "Quick Save" with jQuery



When you are in an administration panel, sometimes you want a "quick save" feature that allows you to save without leaving the page. Here is how to accomplish this with CakePHP and jQuery.



To start, download [jQuery](#) and the [jQuery Form Plugin JavaScript](#). Include them in your view with the JavaScript helper:

PHP:

```
1. $javascript->link(array('jquery', 'form'), false);
```

Include the `RequestHandler` in your controller detect an Ajax save attempt. Also include the JavaScript helper if you haven't already.

PHP:

```
1. var $helpers = array('Javascript');  
2. var $components = array('RequestHandler');
```

Next we want to override our save function with the ajax quick save. Put this right before your `$this->Model->save($this->data)` in your save action.

PHP:

```
1. if ($this->RequestHandler->isAjax()) {  
2.     if ($this->Article->save($this->data)) {  
3.         echo 'success';  
4.     }  
5.     Configure::write('debug', 0);  
6.     $this->autoRender = false;  
7.     exit();  
8. }
```

This detects if the request is ajax, then saves the data. Then it sends back a simple, "success" message to let you know things went fine. It also writes debug to 0 and doesn't render anything, then exits.

Lastly, lets create and include a JavaScript file that performs the quick save.

JavaScript:

```
1. jQuery(function ($) {  
2.     $('<input type="button" value="Quick Save" />')
```

```

3.         .click(function() {
4.             $(this).parents("form:first").ajaxSubmit({
5.                 success: function(responseText, responseCode) {
6.                     $('#ajax-save-
message').hide().html(responseText).fadeIn();
7.                     setTimeout(function() {
8.                         $('#ajax-save-message').fadeOut();
9.                     }, 5000);
10.                }
11.            });
12.            return false;
13.        })
14.        .appendTo('form div.submit');
15.    );

```

This adds a button called, "Quick Save" to each form on the page where a `div` with `class="submit"` exists (you may want to switch this to the id of the form you want to add quick save to). Then It also attaches a click event to the button that submits the form via the [jQuery Form Plugin](#).

In a few simple steps, we've created a quick save feature that saves your data whenever you want without leaving the page.

## 38. PHP & jQuery image upload and crop v1.2

Ensure you have the following:

- PHP 4 or Higher (It has been tested on Version 5)
- PHP GD Library
- [jQuery ver 1.2.3](#) or Higher
- [Image Area Select](#) plugin by Michal Wojciechowski

### BASIC OVERVIEW

1. We will be using the session variable to hold the random file name (in our case the timestamp).

We are now also storing the file extension that is passed through the script, to ensure we are dealing with the right type of image.

1. //only assign a new timestamp if the session variable is empty
2. if (strlen(\$\_SESSION['random\_key'])==0){
3.   \$\_SESSION['random\_key'] = strtotime(date('Y-m-d H:i:s'));//assign the timestamp to the session variable
4.   \$\_SESSION['user\_file\_ext']= "";
5. }

**NOTE:** Once the thumbnail is created we set the session variable to equal nothing, this will then allow a new image to be uploaded along with a new name.

2. Capture, rename and resize the uploaded file.

*The validation section has also been updated and been made more secure by checking the mime type as well as the image extension.*

1. foreach (\$allowed\_image\_types as \$mime\_type => \$ext) {
2.   //loop through the specified image types and if they match the extension then break out
3.   //everything is ok so go and check file size
4.   if(\$file\_ext==\$ext && \$userfile\_type==\$mime\_type){
5.     \$error = "";
6.     break;
7.   }else{
8.     \$error = "Only **". \$image\_ext . "** images accepted for upload";
9.   }
10. }
11. //check if the file size is above the allowed limit
12. if (\$userfile\_size > (\$max\_file\*1048576)) {
13.   \$error.= "Images must be under ".\$max\_file."MB in size";
14. }

3. Same as before, we grab the coordinates using the imgAreaSelect plugin and send the details to the server.

- **x1**, **y1** = coordinates of the top left corner of the initial selection area
- **x2**, **y2** = coordinates of the bottom right corner of the initial selection area
- **width** = crop selection width
- **height** = crop selection height

There are a number of options with this plugin which you can see using the link above. We opted for an aspect ratio of 1:1 (height and width of 100px) along with a preview of what we are actually going to crop.

**NOTE:** Using the php calculation below makes the script that much more dynamic, all you now have to do is set the height and width of your thumbnail and the script will calculate the ratio/preview sizes for you! (Thanks to Long for the tip.)

```
1. $(window).load(function () {  
2.     $('#thumbnail').imgAreaSelect({ aspectRatio: "1:thumb_height/thumb_width", onSelectChange: preview });  
3. });
```

The preview function below, is run as soon as you create your selection. This places the right part of the image into the preview window. The second part of the function populates hidden fields which are later passed to the server.

```
1. function preview(img, selection) {  
2.     var scaleX = /selection.width;  
3.     var scaleY = /selection.height;  
4.  
5.     $('#thumbnail + div > img').css({  
6.         width: Math.round(scaleX * ) + 'px',  
7.         height: Math.round(scaleY * ) + 'px',  
8.         marginLeft: '-' + Math.round(scaleX * selection.x1) + 'px',  
9.         marginTop: '-' + Math.round(scaleY * selection.y1) + 'px'  
10.    });  
11.    $('#x1').val(selection.x1);  
12.    $('#y1').val(selection.y1);  
13.    $('#x2').val(selection.x2);  
14.    $('#y2').val(selection.y2);  
15.    $('#w').val(selection.width);  
16.    $('#h').val(selection.height);  
17. }
```

This function is not really required, but helps by checking to see if the user has made a crop selection. In our case it is a required step. The form is submitted if the values exist, i.e. a selection has been made.

```

1. $(document).ready(function () {
2.     $("#save_thumb").click(function() {
3.         var x1 = $("#x1").val();
4.         var y1 = $("#y1").val();
5.         var x2 = $("#x2").val();
6.         var y2 = $("#y2").val();
7.         var w = $("#w").val();
8.         var h = $("#h").val();
9.         if(x1=="" || y1=="" || x2=="" || y2=="" || w=="" || h==""){
10.             alert("You must make a selection first");
11.             return false;
12.         }else{
13.             return true;
14.         }
15.     });
16. });

```

4. Finally it's time to handle these new coordinates and generate our new cropped thumbnail.

```

1. if (isset($_POST["upload_thumbnail"])){
2.     //Get the new coordinates to crop the image.
3.     $x1 = $_POST["x1"];
4.     $y1 = $_POST["y1"];
5.     $x2 = $_POST["x2"]; // not really required
6.     $y2 = $_POST["y2"]; // not really required
7.     $w = $_POST["w"];
8.     $h = $_POST["h"];
9.     //Scale the image to the 100px by 100px
10.    $scale = $thumb_width/$w;
11.    $cropped = resizeThumbnailImage($thumb_image_location, $large_image_location,$w,$h,$x
1,$y1,$scale);
12.    //Reload the page again to view the thumbnail
13.    header("location:".$_SERVER["PHP_SELF"]);
14.    exit();
15. }

```

Take a look at the demo and download a copy to see the full working code.

So far we have tested it in FireFox 2 and 3, Internet Explorer 6 and 7 and Safari 3.1.2 with pretty good results. The only issue we found was when uploading transparent gifs and pngs results in a black background.

#### [THE DEMO](#)

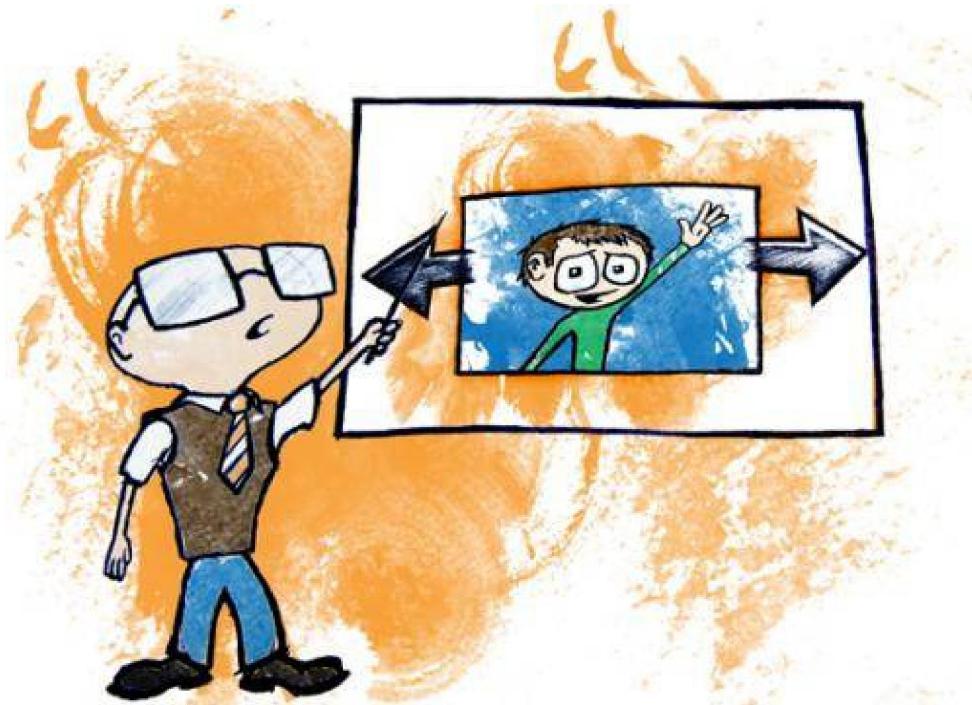
[Click here to see it in action](#)

#### [DOWNLOAD](#)

[Click here to download v1.2](#) [JPG, GIF, PNG]

[Click here to download v1.1](#) [JPG ONLY]

## 39. FlickrScrollr Explained: PHP



### A MORE IN-DEPTH EXPLORATION OF FLICKRSCROLLR

*NOTE: Ok, so I totally [rushed the FlickrScrollr out the door on Tuesday](#) in order to keep the Instant Tip Tuesday series alive (because, let's face it, if I'd only made it three weeks before I missed a Tuesday—well, I wouldn't be much of a blogger at all, would I?). However, in doing so, I didn't really explain how the plugin works or any of the advanced configurations of it. In an effort to rectify that, I'm going to break the code down in this post.*

### THE GOAL OF THIS POST

First, I'd like to readdress [what FlickrScrollr is and how it works](#), then I'd like to pick apart [how the plugin was built](#). Because this plugin uses both a [PHP class](#) and a jQuery plugin, this writeup is going to be a little on the long side, and will be broken into two posts: one for the PHP class, and one to cover the jQuery plugin.

### OVERVIEW

FlickrScrollr is a combination of PHP and jQuery that parses a Flickr RSS feed and creates an animated thumbnail display. It's in its infancy, so please [post any issues you find with it in the comments](#) so I can address them.

[View the Demo](#) | [Download the Source](#)

### REQUIREMENTS

- A server running PHP5+ (SimpleXML won't work on PHP4 or lower)
- The [latest build of jQuery](#)
- A [Flickr account](#)

### IMPLEMENTATION

To implement FlickrScrollr on an HTML page, start by linking to the stylesheets:

```
<link rel="stylesheet" href="css/jquery.lightbox-
0.5.css" type="text/css" media="screen" />
<link rel="stylesheet" href="css/jquery.flickrscrollr.css" type="text/css" me
dia="screen" />
```

Next, include the required JavaScript before the closing body tag:

```
<script type="text/javascript" src="js/jquery-1.3.1.min.js"></script>
<script type="text/javascript" src="js/jquery.lightbox-0.5.min.js"></script>
<script type="text/javascript" src="js/jquery.flickrscrollr-
1.0.1.min.js"></script>
```

With all required stylesheets and scripts included, we parse the feed and generate the markup like so:

```
<?php
    require_once 'inc/FlickrScrollr.inc.php';
    $flickr = new FlickrScrollr('http://api.flickr.com/services/feeds/photos_pu
blic.gne?id=29080075@N02&lang=en-us&format=rss_200');
    echo $flickr;
?>
```

Finally, implement the jQuery plugins to add interactivity:

```
<script type="text/javascript">
    $(document).ready(function() {
        $('#flickrscrollr ul li a').lightBox();
        $('#flickrscrollr').FlickrScrollr();
    });
</script>
```

### CUSTOMIZING FLICKRSCROLLR

FlickrScrollr comes with a few built-in options to allow you to customize it to better suit your needs:

- **thumbWidth**—the width of the thumbnail in pixels plus 1 to accomodate the border. **Default: 76**
- **leftBtn**—the path to the image that will serve as the left button. **Default: 'images/fs\_leftBtn.gif'**
- **rightBtn**—the path to the image that will serve as the right button. **Default: 'images/fs\_rightBtn.gif'**
- **totalPics**—the total number of photos in your RSS feed. Flickr will only provide 20 photos in its stream, so this setting is only for use with sets that contain less than 20 images. It controls the

total width of the scroller, so if there are only ten photos and this is set to 20, the scroller will be too large and will allow the user to scroll beyond your thumbnails. **Default: 20**

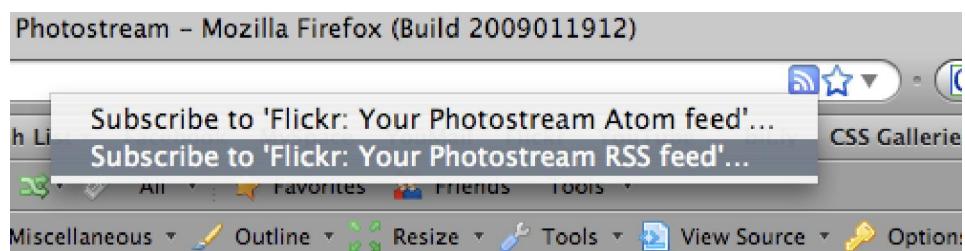
- **displayNum**—the number of photos to display at a time. *NOTE: It's recommended that totalPics is divisible by displayNum to avoid cutting off your display (i.e. totalPics is 20, displayNum should be 1, 2, 4, 5, or 10)* **Default: 4**

To set any of the options to something other than the default value, simply pass them as arguments:

```
$( '#flickrscrollr' ).FlickrScrollr({  
    leftBtn: 'img/newLeftBtn.jpg',  
    rightBtn: 'img/newRightBtn.jpg',  
    displayNum: '5'  
});
```

### UNDERSTANDING FLICKRSCROLLR—STEP 1: THE RSS

To start, we have our RSS feed from Flickr. You find this by going to your photostream, then grabbing the feed's URL, which can be found in two different places: 1) By clicking the RSS icon that appears in the location bar:



Or 2) Clicking the "Latest" link at the bottom of the page:



The URL you end up with should look something like this:

[http://api.flickr.com/services/feeds/photos\\_public.gne?id=29080075@N02&lang=en-us&format=rss\\_200](http://api.flickr.com/services/feeds/photos_public.gne?id=29080075@N02&lang=en-us&format=rss_200)

If you look at the source of the RSS feed, you'll see some XML that looks something like this:

```
<?xml version="1.0" encoding="utf-8"?>  
<rss version="2.0"  
      xmlns:media="http://search.yahoo.com/mrss/"  
      xmlns:dc="http://purl.org/dc/elements/1.1/"  
      xmlns:flickr="urn:flickr:" >  
  
<channel>  
  
    <title>Uploads from jasonatennui</title>
```

```

<link>http://www.flickr.com/photos/ennuidesign/</link>
    <description></description>
    <pubDate>Tue, 11 Nov 2008 11:19:54 -0800</pubDate>

    <lastBuildDate>Tue, 11 Nov 2008 11:19:54 -0800</lastBuildDate>
    <generator>http://www.flickr.com/</generator>
    <image>
        <url>http://farm4.static.flickr.com/3132/buddyicons/29080075@N02.
jpg?1218414199#29080075@N02</url>
        <title>Uploads from jasonatennui</title>
        <link>http://www.flickr.com/photos/ennuidesign/</link>

    </image>

    <item>
        <title>Carly's Senior Photos 01</title>
        <link>http://www.flickr.com/photos/ennuidesign/3023023846/</link>

        <description>&lt;p&gt;&lt;a href="http://www.flickr.com/people/ennuidesign/"&gt;jasonatennui&lt;/a&gt; posted a photo:&lt;/p&gt;
&lt;p&gt;&lt;a href="http://www.flickr.com/photos/ennuidesign/3023023846
" title="Carly's Senior Photos 01"&gt;&lt;img src="http://farm4.static.flickr.com/3235/3023023846_5177ee8057_m.jpg" width="160"
height="240" alt="Carly's Senior Photos 01" /&gt;&lt;/a&gt;&lt;/p&gt;
&lt;p&gt;I took senior photos for my sister. These are my favorites.&lt;/p&gt;
</description>
        <pubDate>Tue, 11 Nov 2008 11:19:54 -0800</pubDate>
        <dc:date.Taken>2008-10-12T12:55:40-08:00</dc:date.Taken>
        <author flickr:profile="http://www.flickr.com/people/ennuidesign/
">nobody@flickr.com (jasonatennui)</author>
        <guid isPermaLink="false">tag:flickr.com,2004:/photo/3023023846</
guid>

        <media:content url="http://farm4.static.flickr.com/3235/302302384
6_55e334f843_o.jpg"
            type="image/jpeg"
            height="3456"
            width="2304"/>
        <media:title>Carly's Senior Photos 01</media:title>
        <media:description type="html">&lt;p&gt;I took senior photos for
my sister. These are my favorites.&lt;/p&gt;</media:description>
        <media:thumbnail url="http://farm4.static.flickr.com/3235/3023023
846_5177ee8057_s.jpg" height="75" width="75" />
        <media:credit role="photographer">jasonatennui</media:credit>
        <media:category scheme="urn:flickr:tags">seniorphotos carlylengst
orf</media:category>

    </item>

    // About 19 more photos go here...

</channel>
</rss>

```

That looks overwhelming, right? Don't worry, we barely have to deal with it at all! The only pieces we need for each photo are these:

```
<title>Carly's Senior Photos 01</title>
<link>http://www.flickr.com/photos/ennuidesign/3023023846/</link>
...
<media:thumbnail url="http://farm4.static.flickr.com/3235/3023023846_5177ee80
57_s.jpg" height="75" width="75" />
```

So how do we sort through all that code to get at the pieces we need? With PHP, it's actually pretty easy!

## STEP 2: THE PHP

In order to handle the RSS feed, we'll be using PHP's SimpleXML extension. We're going to create an object with two methods that will 1) parse the RSS feed and provide the information in an easy-to-use format, and 2) format that information into valid XHTML.

### CREATE THE CLASS

First we need to set up our object and store the feed URL in a variable. We'll do this with a class variable and a [constructor function](#). A constructor is an object-specific method in PHP5 that allows for commands to be executed at the instantiation of a new object. In our case, we want to be able to create a new FlickrScrollr class with the syntax:

```
$obj = new FlickrScrollr('http://api.flickr.com/your_feed_url');
```

However, we also want to allow for the \$feed variable to be directly set, like so:

```
$obj = new FlickrScrollr();
$obj->feed = 'http://api.flickr.com/your_feed_url';
```

By setting the default for \$feed to `NULL` by default, our class doesn't break down if the feed isn't passed on instantiation. Here's what the `__construct()` function looks like:

```
class FlickrScrollr
{
    public $feed;

    function __construct($feed=NULL)
    {
        $this->feed = $feed;
    }
}
```

### PARSE THE FEED

Now we're ready to actually parse the feed. To do this, we're going to create a protected method called `parse()`. This will break the RSS feed into just the pieces we need, and return that information in an easy-to-use array:

```

protected function parse()
{
    $rss = simplexml_load_file($this->feed);
    foreach ($rss->channel->item as $item) {
        $link = $item->link;
        $title = $item->title;
        $media = $item->children('http://search.yahoo.com/mrss/');
        $thumb = $media->thumbnail->attributes();
        $thumbSRC = $thumb['url'];
        $width = $thumb['width'];
        $height = $thumb['height'];
        $imgSRC = str_replace('_s', '', $thumbSRC);

        $photos[] = array($link, $title, $thumbSRC, $imgSRC, $width, $height)
    }
}

return $photos;
}

```

So, let's take this in baby steps. First, we load the RSS feed using SimpleXML. This puts all of the information in the feed into one variable, `$rss`. We then pass that variable to a `foreach` loop, but we drill a little deeper into the variable to extract each photo. The way SimpleXML works, you can move through the XML in a very intuitive fashion: simply connect the dots to get to the information you want!

Passing `$rss->channel->item as $item` effectively chunks the RSS feed into individual entries. In the case of our Flickr feed, each photo is contained between `<item>` tags (this is a standard in RSS feeds).

Once the photos have been separated, we just continue drilling to access the photo's URL and title:

```

$link = $item->link;
$title = $item->title;

```

Ah, but wait! There's some tomfoolery going on with our thumbnail that takes a little further explanation. What's with the `<:media:thumbnail ... />`?

The "media:thumbnail" is an example of an XML concept called "namespaces"—essentially, a namespace in XML is a way to guarantee that an XML value is unique. If you look toward the top of the RSS feed, you'll see this line:

```

<rss version="2.0"
    xmlns:media="http://search.yahoo.com/mrss/"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:flickr="urn:flickr:" >

```

In effect, `xmlns:media="http://search.yahoo.com/mrss/"` is creating an alias for any XML tag started with `media`:—every tag will be replaced with the URL, which guarantees that the tag is unique, no matter what other XML file it may be combined with.

What that means to us, as developers, is that we need to access the namespace in order to use the information it stores. Fortunately, this is supported by SimpleXML. To access the `media`: namespace, we use the following snippet of code:

```
$media = $item->children('http://search.yahoo.com/mrss/');
```

Now all the XML tags containing the `media`: namespace will be stored in the `$media` variable, in the same way the whole feed was stored in `$rss`. Now we access the thumbnail, but we have to use the `attributes` call, since the information is stored as an attribute in the tag, rather than as a value (i.e. `<tag>value</tag>`).

```
$thumb = $media->thumbnail->attributes();
$thumbSRC = $thumb['url'];
$width = $thumb['width'];
$height = $thumb['height'];
```

Finally, we need to get the URL of the medium-size image Flickr creates. Strangely, this is the only URL that isn't included in the feed, so we'll have to employ a little creativity to get it. The way Flickr names their images, they append a size identifier to the end of the image number (i.e. `_s` for square, or thumbnail, `_o` for original size, etc.). The medium-size image is the default, so nothing is appended. Therefore, to get to it, we can just pop the `_s` off our thumbnail URL, like so:

```
$imgSRC = str_replace('_s', '', $thumbSRC);
```

Finally, we just return all the information we've grabbed into a multi-dimensional array.

### EXTRACT THE USER ID

The next method, `getUserId()` is a little tricky, as we're using a [regular expression](#). What we're attempting to do is to extract the user ID out of the feed URL to allow us to link directly to the user's photostream. Originally, I had simply used a series of `str_replace()` functions, but it was pointed out by [Will](#) that not all Flickr RSS feeds are formatted the same, meaning some feeds wouldn't match the strings I'd defined, leaving room for errors and/or bad links.

A regular expression allows for pattern matching, rather than specific string matching, which is both very useful and a little intimidating. Consider our pattern:

```
$pattern = "/^(.+id=) ([A-Za-z0-9@]{1,}) (&.*)$/";
```

I won't go into the fine details of regex here (that's an entire blog post/series in and of itself), but what we're essentially saying in this pattern is, "From the beginning of the supplied string, match everything up until you find the characters 'id='), then match one or more letters (both upper- and lowercase) or digits until the string meets an ampersand (&), then grab everything up until the end of the string."

Each piece of the pattern enclosed within parentheses (i.e. "(.\*id-)") will be stored in a special variable, which is simply a sequentially numbered PHP variable (i.e. \$1, \$2, \$3, etc.). We can use these special variables to reformat the matched elements as we see fit. For our purposes, we just want to pull the second match (the information between 'id=' and the next ampersand), since that's our user ID. We then pass this pattern and replacement, along with the feed URL, to `preg_replace()`, which looks like this:

```
$replacement = '$2';
return preg_replace($pattern, $replacement, $this->feed);
```

With our user ID now stored in a variable, we're ready to build our markup!

#### PUT TOGETHER THE XHTML MARKUP

Our next method is going to be a public one called `display()`:

```
public function display()
{
    $thumbs = NULL;
    $pics = $this->parse();
    $user = $this->getUserId();

    foreach($pics as $pic) {
        $thumbs .= <<<THUMBNAIL_DISPLAY

            <li>
                <a href="{{$pic[3]}}" title=<a href="{{$pic[0]}}>{{$pic[1]}}</a>">
                    
                </a>
            </li>
        THUMBNAIL_DISPLAY;
    }

    $display = <<<FLICKR_DISPLAY

<div id="flickrscrollr">
    <div id="fs_wrapper">
        <ul>$thumbs
        </ul>
    </div>
    <a href="http://flickr.com/photos/$user/" class="fs_morelink">View All Photos</a>
</div>
FLICKR_DISPLAY;

    return $display;
}
```

Our first step is to declare an empty variable that will hold the thumbnails as we step through the array returned by `parse()`, which is our next step. We load the return value of `parse()` into a

variable we'll name \$pics, and load the user ID extracted by `getUserId()` into a variable called \$user

Next we initiate a loop to cycle through our array of photos and build list items, using the array index of each piece of information returned by `parse()` to place the necessary information into the markup.

```
foreach($pics as $pic) {  
    $thumbs .= <<<THUMBNAIL_DISPLAY  
  
    <li>  
        <a href="{{$pic[3]}}" title=<a href="{{$pic[0]}}>{{$pic[1]}}</a>"  
>  
              
        </a>  
    </li>  
THUMBNAIL_DISPLAY;  
}
```

After cycling through all of the photos, we place the markup into a wrapper, leaving it ready for display:

```
$display = <<<FLICKR_DISPLAY  
  
<div id="flickrscrollr">  
    <div id="fs_wrapper">  
        <ul>$thumbs  
        </ul>  
    </div>  
    <a href="http://flickr.com/photos/$user/" class="fs_morelink">View All Photos</a>  
    </div>  
FLICKR_DISPLAY;  
  
    return $display;
```

#### FINISHING TOUCHES

Finally, we use the magic method `__toString()` to allow us to output the contents of `display()` simply by echoing the `FlickrScrollr` object:

```
public function __toString()  
{  
    return $this->display();  
}
```

This step is a precaution against errors caused by trying to echo an object without a `__toString()` method declared. If this method is omitted, attempting to echo the object (i.e. `echo $flickrscrollr;`) would cause a fatal error.

## 40. PHP Login script

Learn to create a simple login system with php + mysql script, this tutorial easy to follow, teach you step by step.

### Overview

In this tutorial create 3 files

1. main\_login.php
2. checklogin.php
3. login\_success.php

### Step

1. Create table "members" in database "test".
2. Create file main\_login.php.
3. Create file checklogin.php.
4. Create file login\_success.php.
5. Create file logout.php

[If you don't know how to create database, click here](#)



### Create table "members"

```
CREATE TABLE `members` (
`id` int(4) NOT NULL auto_increment,
`username` varchar(65) NOT NULL default '',
`password` varchar(65) NOT NULL default '',
PRIMARY KEY (`id`)
) TYPE=MyISAM AUTO_INCREMENT=2 ;
```

--

-- Dumping data for table `members`

--

```
INSERT INTO `members` VALUES (1, 'john', '1234');
```

Database "test"			
Table "members"			
	id	username	password
	1	john	1234

## ② Create file main\_login.php

[View In Browser](#)

Member Login	
Username :	<input name="myusername" type="text"/>
Password :	<input name="mypassword" type="text"/>
<input type="submit" value="Login"/> <input name="action" type="hidden" value="checklogin.php"/>	

### ##### Code

```
<table width="300" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#CCCCCC">
<tr>
<form name="form1" method="post" action="checklogin.php">
<td>
<table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="#FFFFFF">
<tr>
<td colspan="3"><strong>Member Login </strong></td>
</tr>
<tr>
<td width="78">Username</td>
<td width="6">:</td>
<td width="294"><input name="myusername" type="text" id="myusername"></td>
</tr>
<tr>
<td>Password</td>
```

```

<td></td>
<td><input name="mypassword" type="text" id="mypassword"></td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td><input type="submit" name="Submit" value="Login"></td>
</tr>
</table>
</td>
</form>
</tr>
</table>

```

### Create file checklogin.php

#### ##### Code

```

<?php
$host="localhost"; // Host name
$username=""; // Mysql username
$password=""; // Mysql password
$db_name="test"; // Database name
$tbl_name="members"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// username and password sent from form
$myusername=$_POST['myusername'];
$mypassword=$_POST['mypassword'];

// To protect MySQL injection (more detail about MySQL injection)
$myusername = stripslashes($myusername);
$mypassword = stripslashes($mypassword);
$myusername = mysql_real_escape_string($myusername);

```

```

$mypassword = mysql_real_escape_string($mypassword);

$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
password='$mypassword'";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);
// If result matched $myusername and $mypassword, table row must be 1 row

if($count==1){
// Register $myusername, $mypassword and redirect to file "login_success.php"
session_register("myusername");
session_register("mypassword");
header("location:login_success.php");
}
else {
echo "Wrong Username or Password";
}
?>

```

#### Create file login\_success.php

#### ##### Code

```

// Check if session is not registered , redirect back to main page.
// Put this code in first line of web page.
<?
session_start();
if(!session_is_registered(myusername)){
header("location:main_login.php");
}
?>

<html>
<body>
Login Successful
</body>
</html>

```

## 5 Logout.php

If you want to logout, create this file

```
// Put this code in first line of web page.  
<?  
session_start();  
session_destroy();  
?>
```

## 6 For PHP5 User - checklogin.php

##### Code

```
<?php  
ob_start();  
$host="localhost"; // Host name  
$username=""; // Mysql username  
$password=""; // Mysql password  
$db_name="test"; // Database name  
$tbl_name="members"; // Table name  
  
// Connect to server and select database.  
mysql_connect("$host", "$username", "$password")or die("cannot connect");  
mysql_select_db("$db_name")or die("cannot select DB");  
  
// Define $myusername and $mypassword  
$myusername=$_POST['myusername'];  
$mypassword=$_POST['mypassword'];  
  
// To protect MySQL injection (more detail about MySQL injection)  
$myusername = stripslashes($myusername);  
$mypassword = stripslashes($mypassword);  
$myusername = mysql_real_escape_string($myusername);  
$mypassword = mysql_real_escape_string($mypassword);  
  
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and
```

```

password='$mypassword";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);
// If result matched $myusername and $mypassword, table row must be 1 row

if($count==1){
// Register $myusername, $mypassword and redirect to file "login_success.php"
session_register("myusername");
session_register("mypassword");
header("location:login_success.php");
}
else {
echo "Wrong Username or Password";
}

ob_end_flush();
?>

```



## 7 Encrypting Password - Make your Login More Secure

**This is the same password**

<b>id</b>	<b>name</b>	<b>email</b>	<b>password</b>
1	John Smith	john@somewhere.com	john856

<b>id</b>	<b>name</b>	<b>email</b>	<b>password</b>
1	John Smith	john@somewhere.com	ad65d5054042fda44ba3fdc97cee80c6

**After encrypted "john856"**

[Read more about encrypting password here](#)

## 41. Tips Membuat Script PHP Pengolah Password dengan MD5

Setiap password pengguna aplikasi hendaknya dienkripsi untuk keperluan faktor keamanan. Mengapa password harus dienkripsi? atau apa sih enkripsi itu? OK... enkripsi adalah teknik penyandian pesan, yang semula pesan tersebut dapat dibaca dan bermakna, setelah dienkripsi menjadi tidak terbaca dan tidak bermakna. Lantas... mengapa password harus dienkripsi? Pertanyaan tersebut saya balik, bagaimana jika password tidak dienkripsi? Apabila password tidak dienkripsi, maka dapat dengan mudah dibaca, dan digunakan oleh orang lain yang tidak berhak untuk masuk ke dalam sistem atau aplikasi.

Dalam PHP, tentu kita tidak asing dengan perintah atau function md5(). Function ini sering digunakan para programmer untuk mengenkripsi password sebelum hasil enkripsi tersebut disimpan dalam database sistem, ketika registrasi user baru. Hasil enkripsi md5() berupa suatu string acak dengan panjang 32 karakter (256 bit). Sudah amankah penggunaan md5()? Artikel ini akan membahasnya, serta memberikan tips bagaimana cara membuat script PHP yang baik untuk mengolah password.

Nah... biasanya, struktur code untuk mengenkripsi password dengan md5 dan menyimpannya ke dalam database (pada registrasi user baru) adalah sebagai berikut:

```
1 <?php
2
3 // input username baru
4 // input password asli baru ($passAsli)
5
6 $passEnkrip = md5($passAsli);
7
8 /* procedure simpan data username
9 dan password hasil md5() ke db */
10
11 ?>
```

Sedangkan struktur code untuk loginnya adalah

```
1 <?php
2
3 // input login username
4 // input login password asli ($passAsli)
5
6 if (md5($passAsli) == $passEnkrip)
7 {
8     // login sukses
9     // procedure jika login sukses
10 }
11 else {
12     // login gagal
13     // procedure jika login gagal
14 }
15
```

16 ?>

Lantas... benar-benar sudah amankah penggunaan md5() dengan struktur code di atas?

Beberapa periode yang lalu, mungkin penggunaan struktur seperti di atas sudah dirasa aman. Namun saat ini tidak aman lagi, karena sudah banyak tool untuk mendekripsi hasil enkripsi md5(). Salah satu toolnya seperti yang ada di situs <http://md5.rednoize.com><sup>[1]</sup>. Apa akibatnya jika password ini didekripsi? wah bahaya... bisa-bisa password aslinya ketahuan.

So... gimana donk? apakah md5() tidak usah digunakan lagi? Tidak usah khawatir, kita tetap bisa menggunakan md5() namun perlu sedikit kreatif. Maksudnya adalah bahwa kita perlu mengkombinasikan penggunaan md5() dengan pengacak, misalnya kita gunakan md5() berulangkali, atau menggabungkan password asli dengan suatu string tertentu lalu dienkripsi.

Berikut ini contoh struktur code untuk menyimpan password terenkripsi menggunakan pengacak

```
1 <?php
2
3 // input username baru
4 // input password baru ($passAsli)
5
6 $pengacak = "AJWKXLAJSCLWLW";
7 $passEnkrip = md5($pengacak . md5($passAsli) . $pengacak );
8
9 /* procedure simpan data username
10   dan password $passEnkrip ke db */
11
12 ?>
```

Sedangkan berikut ini adalah struktur untuk loginnya

```
1 <?php
2
3 // input login username
4 // input login password asli ($passAsli)
5
6 $pengacak = "AJWKXLAJSCLWLW";
7
8 if (md5($pengacak . md5($passAsli) . $pengacak) == $passEnkrip)
9 {
10    // login sukses
11    // procedure jika login sukses
12 }
13 else {
14    // login gagal
15    // procedure jika login gagal
16 }
17 ?>
```

Anda dapat mengubah isi pengacak atau mungkin mengubah format enkripsinya menjadi model lain, misalnya menggabungkan 3 atau lebih md5() dalam enkripsi. Dalam hal ini, hanya

kita yang tahu format enkripsi atau pengacaknya. Intinya adalah jangan mengenkripsi password menggunakan md5() secara langsung, karena hal ini rawan untuk dihack pada saat ini.

Mudah-mudahan artikel ini membantu...

## 42. Tentang Letak Script Pengolah Form

Misalkan Anda memiliki sebuah form untuk entri input. Trus... input tersebut akan Anda proses dimana? Apakah harus di script yang berbeda dengan formnya? Biasanya memang script pengolah input ini diletakkan tidak satu tempat dengan form inputnya. Namun... ini tidak merupakan satu-satunya cara untuk meletakkan script pengolah form. Script pengolah form ini dapat Anda letakkan dalam script yang sama dengan formnya.

Bagaimana cara melakukannya? Simak artikel berikut ini lebih lanjut.

OK... contoh sederhana misalkan kita akan membuat kalkulator sederhana untuk menghitung penjumlahan dua buah bilangan dengan script PHP.

Berikut ini adalah kode HTML untuk membuat form input bilangannya.

```
<form method="post" action="...">
    <table>
        <tr><td>Bil Pertama</td><td>:</td><td><input type="text"
name="bil1"></td></tr>
        <tr><td>Bil Kedua</td><td>:</td><td><input type="text"
name="bil2"></td></tr>
        <tr><td></td><td></td><td><input type="submit" name="submit"
value="Submit"></td></tr>
    </table>
</form>
```

Nah.. biasanya kita mengarahkan script PHP pengolah form di atas ke script yang berbeda dengan script formnya. Sekarang akan kita coba untuk mengarahkan form tersebut ke script yang jadi satu dengan formnya atau ke dirinya sendiri.

Untuk mengarahkan submit form ke script itu sendiri, gunakan perintah `$_SERVER['PHP_SELF']`.

```
<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
    <table>
        <tr><td>Bil Pertama</td><td>:</td><td><input type="text"
name="bil1"></td></tr>
        <tr><td>Bil Kedua</td><td>:</td><td><input type="text"
name="bil2"></td></tr>
        <tr><td></td><td></td><td><input type="submit" name="submit"
value="Submit"></td></tr>
    </table>
</form>
```

Kemudian, dimanakah kita letakkan script pengolah scriptnya? Ya... ini terserah kita. Bisa kita letakkan di atas form atau di bawah form. Contoh berikut ini adalah menampilkan hasil penjumlahan di atas form.

```

<?php
    $bill1 = $_POST['bill'];
    $bill2 = $_POST['bil2'];
    $hasil = $bill1 + $bill2;
    echo "Hasil penjumlahannya adalah ".$hasil;
?>

<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
    <table>
        <tr><td>Bil Pertama</td><td>:</td><td><input type="text"
name="bill1"></td></tr>
        <tr><td>Bil Kedua</td><td>:</td><td><input type="text"
name="bil2"></td></tr>
        <tr><td></td><td><td><input type="submit" name="submit"
value="Submit"></td></tr>
    </table>
</form>

```

Lalu.. bagaimana seandainya kita ingin bahwa yang ditampilkan hanya hasil penjumlahannya saja dan formnya tidak muncul, akan tetapi scriptnya jadi satu dalam file yang sama? Tentu ini bisa dilakukan. Konsepnya adalah kita menggunakan statement IF. IF ini kita gunakan untuk mengecek apakah tombol submit ini diklik atau tidak. Jika diklik maka tampilkan hasilnya. Sedangkan bila tidak diklik maka tampilkan formnya.

Untuk mengecek apakah tombol submit diklik atau tidak, gunakan perintah

```

if ($_POST['submit'])
{
    // tampilkan hasil penjumlahan
}
else
{
    // tampilkan form nya
}

```

Sehingga script di atas menjadi seperti di bawah ini

```

<?php

if ($_POST['submit'])
{
    $bill1 = $_POST['bill'];
    $bill2 = $_POST['bil2'];
    $hasil = $bill1 + $bill2;
    echo "Hasil penjumlahannya adalah ".$hasil;
}
else
{
?>

<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
    <table>
        <tr><td>Bil Pertama</td><td>:</td><td><input type="text"
name="bill1"></td></tr>

```

```

<tr><td>Bil Kedua</td><td>:</td><td><input type="text"
name="bil2"></td></tr>
<tr><td></td><td></td><td><input type="submit" name="submit"
value="Submit"></td></tr>
</table>
</form>

<?php
}
?>
```

Selain cara yang kedua diatas, kita juga bisa mengarahkan input formnya ke script itu sendiri tapi menggunakan parameter. Berikut ini contohnya:

```

<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>?aksi=proses">
<table>
    <tr><td>Bil Pertama</td><td>:</td><td><input type="text"
name="bill1"></td></tr>
    <tr><td>Bil Kedua</td><td>:</td><td><input type="text"
name="bil2"></td></tr>
    <tr><td></td><td></td><td><input type="submit" name="submit"
value="Submit"></td></tr>
</table>
</form>
```

Pada form di atas, data input diarahkan ke script dia sendiri tapi dengan parameter ?action=proses. Untuk cara ini, kita tidak usah mendeteksi klik submitnya seperti pada contoh kedua di atas. Trus... bagaimana cara menghandlenya? Caranya hampir sama dengan mendeteksi klik tombol submit, namun dalam hal ini kita gunakan logika seperti ini:

```

if ($_GET['aksi'] == "proses")
{
    // tampilkan hasil penjumlahan
}
else
{
    // tampilkan form input
}
```

Mengapa menggunakan \$\_GET['aksi']? Ya... karena dalam action terdapat parameter 'aksi'. Parameter ini juga merupakan submit yang menggunakan method GET. Sedangkan 'proses' merupakan value dari parameter 'aksi' ini. Sehingga untuk mendeteksi apakah input sudah disubmit adalah hanya mengecek nilai dari parameter 'aksi' ini. Jika nilai parameter 'aksi' = 'proses' maka tampilkan hasil penjumlahan. Bila tidak, maka tampilkan formnya.

Dengan demikian script cara ketiga untuk mengolah input menggunakan action berbentuk parameter adalah sbb:

```

<?php
if ($_GET['aksi'] == "proses")
{
    $bill1 = $_POST['bill1'];
}
```

```

$bil2 = $_POST['bil2'];
$hasil = $bil1 + $bil2;
echo "Hasil penjumlahannya adalah ".$hasil;
}
else
{
?>
<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>?aksi=proses">


```

So... enaknya mau meletakkan script pengolah form ini dimana? di dalam script yang sama dengan formnya atau di script yang lain? It's up to you.. Semuanya terserah Anda, karena Anda sang programmer.

Bagaimana bila tombol submitnya lebih dari satu, seperti yang pernah saya bahas di artikel [Beberapa Tombol Submit dalam Satu Form](#)<sup>[1]</sup>? Bagaimana cara membuatnya bila input diarahkan ke script yang sama dengan formnya? Ya... caranya hampir sama, nanti Anda gunakan logika seperti ini

```

if ($_POST['submit1'])
{
    // action apabila tombol submit 1 diklik
}
else if ($_POST['submit2'])
{
    // action apabila tombol submit 2 diklik
}
else if ($_POST['submit3'])
{
    // action apabila tombol submit 3 diklik
}
.
.
.
else
{
    // action bila tidak ada tombol yang diklik
    // menampilkan form inputnya
    // jangan lupa gunakan action="<?php echo $_SERVER['PHP_SELF']; ?>" pada
formnya
}

```

### 43. Membuat Tampilan Data Terurut Berdasarkan Kolom Tabel

Waduh.. lama tidak menyapa pengunjung setia [blog ini](#)<sup>[1]</sup> melalui sharing ilmunya ya? Maaf ya karena alasan kesibukan di kantor dan mengerjakan beberapa project pribadi, blog ini lama gak diupdate dengan artikel-artikel terbaru.

OK deh, saya kali ini akan membahas bagaimana cara membuat tampilan data terurut berdasarkan kolom tabel. Agak bingung juga nulis judul artikel ini. Tapi... secara garis besar maksudnya sebagaimana tampak pada gambar di bawah ini.

## Data Mahasiswa

<u>NIM</u>	<u>Nama Mhs</u>	<u>Alamat</u>	<u>Jenis Kelamin</u>	<u>Tgl Lahir</u>
M0001	A	SOLO	L	1979-10-20
M0002	D	SOLO	L	1981-10-12
M0003	B	BANDUNG	P	1980-11-07
M0004	C	JAKARTA	P	1980-09-18

Andaikan kita punya data dalam database, kemudian kita ingin menampilkan data-data tersebut ke dalam halaman web seperti di atas. Dalam studi kasus ini, saya gunakan data mahasiswa. Nah.. secara default data yang tampil dalam tabel di atas akan terurut berdasarkan NIM nya. Tidak hanya NIM saja yang bisa digunakan sebagai dasar pengurutan data, namun juga field yang lain misalnya NAMA, ALAMAT, JENIS KELAMIN atau bahkan TANGGAL LAHIR nya. Untuk mengurutkan data yang tampil berdasarkan field tertentu cukup dengan mengklik nama kolomnya saja.

Perhatikan gambar berikut ini yang menunjukkan tampilan data apabila diklik pada kolom NAMA MHS (diurutkan berdasarkan nama mahasiswa)

## Data Mahasiswa

<u>NIM</u>	<u>Nama Mhs</u>	<u>Alamat</u>	<u>Jenis Kelamin</u>	<u>Tgl Lahir</u>
M0001	A	SOLO	L	1979-10-20
M0003	B	BANDUNG	P	1980-11-07
M0004	C	JAKARTA	P	1980-09-18
M0002	D	SOLO	L	1981-10-12

Demikian pula gambar berikut ini yang menunjukkan tampilan data bila diklik pada kolom TANGGAL LAHIR (diurutkan berdasarkan tanggal lahir)

## Data Mahasiswa

NIM	Nama Mhs	Alamat	Jenis Kelamin	Tgl Lahir
M0001	A	SOLO	L	1979-10-20
M0004	C	JAKARTA	P	1980-09-18
M0003	B	BANDUNG	P	1980-11-07
M0002	D	SOLO	L	1981-10-12

OK... paham ya maksudnya membuat tampilan data terurut berdasarkan kolom tabel?

Now... akan saya jelaskan cara membuatnya. Pertama, tentu saja kita harus sudah memiliki data yang mau ditampilkan dahulu. Di sini saya menggunakan tabel ‘mahasiswa’ dengan struktur sbb:

```
CREATE TABLE `mahasiswa` (
  `nim` varchar(5),
  `nama` varchar(20),
  `alamat` text,
  `sex` varchar(1),
  `tgllahir` date,
  PRIMARY KEY (`nim`)
);
```

serta beberapa buah data sebagai sampelnya.

```
INSERT INTO `mahasiswa` VALUES ('M0001', 'A', 'SOLO', 'L', '1979-10-20');
INSERT INTO `mahasiswa` VALUES ('M0003', 'B', 'BANDUNG', 'P', '1980-11-07');
INSERT INTO `mahasiswa` VALUES ('M0002', 'D', 'SOLO', 'L', '1981-10-12');
INSERT INTO `mahasiswa` VALUES ('M0004', 'C', 'JAKARTA', 'P', '1980-09-18');
```

OK sekarang kita masuk ke tahap berikutnya yaitu proses membuat scriptnya. Seperti biasa sebelum kita mulai membuat script, kita harus punya konsep atau ide penyelesaiannya dahulu. Konsepnya adalah kita gunakan statement IF atau kondisional. Why? ya... karena tampilan datanya juga kondisional atau bersyarat. Mengapa bersyarat? OK.. ini dia alasannya. Kapan tampilan data yang muncul terurut berdasarkan NAMA? ya... ketika kolom NAMA MHS nya diklik mas? OK.. benar.. trus, kapan tampilan data yang muncul terurut berdasarkan TANGGAL LAHIR nya? ketika kolom TANGGAL LAHIR nya diklik donk? that's right!!

Nah.. secara umum, ketika kolom NIM diklik maka script akan menjalankan query berikut ini untuk menampilkan datanya

```
SELECT * FROM mahasiswa ORDER BY nim
```

Sedangkan bila kolom NAMA MHS diklik, maka query yang dijalankan adalah

```
SELECT * FROM mahasiswa ORDER BY nama
```

Bila kolom ALAMAT diklik, maka query yang dijalankan adalah

```
SELECT * FROM mahasiswa ORDER BY alamat
```

Begitu seterusnya..

OK.. konsep dan ide dah dapat, so... kita bisa action sekarang. Ini dia scriptnya.

```
<h1>Data Mahasiswa</h1>

<!-- membuat kolom tabel dan link pengurutan -->
<!-- perhatikan parameter by=... -->



| <a href="&lt;?php \$_SERVER['PHP_SELF'] ?&gt;?by=nim">NIM</a> | <a href="&lt;?php \$_SERVER['PHP_SELF'] ?&gt;?by=nama">Nama Mhs</a> | <a href="&lt;?php \$_SERVER['PHP_SELF'] ?&gt;?by=alamat">Alamat</a> | <a href="&lt;?php \$_SERVER['PHP_SELF'] ?&gt;?by=sex">Jenis Kelamin</a> | <a href="&lt;?php \$_SERVER['PHP_SELF'] ?&gt;?by=tgl">Tgl Lahir</a> |
|---------------------------------------------------------------|---------------------------------------------------------------------|---------------------------------------------------------------------|-------------------------------------------------------------------------|---------------------------------------------------------------------|
|---------------------------------------------------------------|---------------------------------------------------------------------|---------------------------------------------------------------------|-------------------------------------------------------------------------|---------------------------------------------------------------------|


```

```
?>  
</table>
```

Catatan:

Perhatikan script di atas, bahwa untuk mendeteksi bahwa kolom apa yang diklik digunakanlah `$_GET['by']` untuk membaca nilai parameter yang ada di setiap link nama kolomnya.

Yes!! Mudah bukan membuat script PHP untuk tampilan data terurut berdasarkan kolom tabel? Nah... silakan bereksperimen dan mengembangkan scriptnya sendiri ya. Mungkin Anda bisa tambahkan beberapa fitur tambahan misalnya sortingnya secara ASCENDING atau DESCENDING. OK.. selamat mencoba.

## 44. Teknik Membuat On Page Paging

Seringkali apabila kita menjumpai teknik tentang paging, paging yang dimaksud terkait dengan cara penyajian sejumlah data ke dalam beberapa halaman. Saya pun pernah membahas teknik paging ini pada salah satu artikel saya tentang [advanced paging](#)<sup>[1]</sup>.

Namun.. bagaimana bila teknik paging ini diterapkan pada sebuah artikel? Idenya begini, misalkan kita membuat script CMS untuk menampilkan sebuah artikel. Selanjutnya andaikan kita punya sebuah artikel yang suangaa..aat panjang. Karena saking panjangnya, kita ingin membuat artikel tersebut terbagi ke dalam beberapa halaman. Nah... teknik ini nantinya juga akan menggunakan paging.

Trus.. bagaimana tuh cara membuatnya?

OK... sebagai contoh, misalkan kita memiliki tabel untuk menyimpan artikel seperti di bawah ini:

```
CREATE TABLE `post` (
  `id` int(11) AUTO_INCREMENT,
  `title` varchar(100),
  `konten` text,
  PRIMARY KEY (`id`)
)
```

Untuk tabel contoh di atas, sengaja saya ambil sederhana saja ya, yaitu terdiri dari field ‘id’, ‘title’, dan ‘konten’. Anda bisa menambahkan field yang lain, misalnya tanggal posting, author dan juga kategori artikel.

Sedangkan contoh data artikelnya sebagaimana tersaji pada contoh berikut ini:

```
INSERT INTO `post` VALUES ('1', 'Judul 1', 'ini adalah halaman pertama. ini  
adalah halaman pertama. ini adalah halaman pertama. ini adalah halaman  
pertama. ini adalah halaman pertama. ini adalah halaman pertama.<!--next-->  
ini adalah halaman kedua. ini adalah halaman kedua. ini adalah halaman  
kedua. ini adalah halaman kedua. ini adalah halaman kedua. ini adalah halaman  
kedua. ini adalah halaman kedua. ini adalah halaman kedua. ini adalah halaman  
kedua. ini adalah halaman kedua. ini adalah halaman ketiga. ini  
adalah halaman ketiga. ini adalah halaman ketiga. ini adalah halaman ketiga.  
ini adalah halaman ketiga. ini adalah halaman ketiga. ini adalah halaman  
ketiga. ');
INSERT INTO `post` VALUES ('2', 'Judul 2', 'ini adalah halaman pertama. ini  
adalah halaman pertama. ini adalah halaman pertama. ini adalah halaman  
pertama. ini adalah halaman pertama. ini adalah halaman pertama.<!--next-->  
ini adalah halaman kedua. ini adalah halaman kedua. ini adalah halaman  
kedua. ini adalah halaman kedua. ini adalah halaman kedua. ini adalah halaman  
kedua. ini adalah halaman kedua. ini adalah halaman ketiga. ini  
adalah halaman ketiga. ini adalah halaman ketiga. ini adalah halaman ketiga.  
ini adalah halaman ketiga. ini adalah halaman ketiga. ini adalah halaman  
ketiga.');
```

Bila Anda perhatikan sampel data di atas, terlihat bahwa di dalam postingan terdapat tanda

```
<!--next-->
```

Tanda ini nantinya digunakan untuk pemisah halaman pada artikel tersebut. Konsep penggunaan tanda pemisah ini saya adopsi dari teknik yang digunakan oleh Wordpress. Tanda itu sebaiknya disisipkan secara manual ke dalam artikel, bukannya secara otomatis, karena hanya author atau penulis artikelah yang tahu kapan dan di bagian mana harus memenggal halamannya.

Oya... Anda juga boleh menggunakan atau membuat tanda pemisah lain sesuai keinginan sendiri.

OK... kita lanjutkan. Sekarang untuk skenarionya, pertama kita buat script untuk memunculkan semua judul artikel yang ada dalam tabel database. Selanjutnya bila kita klik salah satu title, maka akan tampil full artikelnya.

Berikut ini adalah script untuk memunculkan semua title artikelnya.

### artikel.php

```
<?php
mysql_connect ("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

$query = "SELECT * FROM post";
$hasil = mysql_query($query);

// menampilkan list judul artikel

echo "<ul>";
while ($data = mysql_fetch_array($hasil))
{
    echo "<li><a href='view.php?id=".$data['id']."'>".$data['title']."</a></li>";
}
echo "</ul>";
?>
```

Hasil dari script di atas tampak pada gambar di bawah ini

- Judul 1
- Judul 2



A screenshot of a web browser window. The address bar at the top shows the URL `http://localhost/view.php?id=1`. Below the address bar, there is a list of two items, each preceded by a bullet point. The first item is Judul 1, and the second item is Judul 2. Both items are underlined, indicating they are links.

Nah.. kemudian kita akan membuat script view.php nya. Namun sebelum kita mulai membuatnya, kita harus punya ide terlebih dahulu.

Bagaimana idenya? OK misalkan kita punya sebuah string sbb:

```
xxxxxx<!--next-->yyyyyy<!--next-->zzzzzz
```

Untuk konsep paging on page tersebut, langkah pertama kita pecah dahulu string di atas ke dalam beberapa substring. Substring ini nanti akan dikaitkan dengan masing-masing halaman. Pemecahan tersebut dilakukan berdasar tanda . Sehingga dari proses ini kita dapatkan 3 buah substring (3 buah halaman) yang tersimpan dalam bentuk array, yaitu substring pertama (array ke-0) yang nanti sebagai halaman pertama yaitu ‘xxxxxx’ , substring kedua (array ke-1) yang nanti sebagai halaman kedua, dan substring ketiga (array ke-2) yang nanti sebagai halaman ketiga.

OK... berikutnya kita bisa terapkan konsep di atas pada script view.php

### **view.php**

```
<?php
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// membaca id artikel dari link
$id = $_GET['id'];

// query untuk membaca data artikel berdasar id
$query = "SELECT * FROM post WHERE id = '$id'";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

// memecah artikel berdasarkan tanda pemisah
$pecah = explode("<!--next-->", $data['konten']);

// menampilkan title
echo "<h1>".$data['title']."</h1>";

// menampilkan halaman pertama dari artikel
echo "<p>".$pecah[0]."</p>";

// menampilkan link paging
echo "<p>Halaman: ";
for ($i=1; $i<=count($pecah); $i++)
{
    echo "<a href='view.php?id=".$data['id']."'>".$i."</a> ";
}
echo "<p>";
?

?>
```

Bila kita perhatikan link paging di atas, maka link tersebut berbentuk

```
<a href='view.php?id=...&page=... '>...</a>
```

yang dalam hal ini, nomor halaman yang dituju adalah berdasarkan parameter ‘page’ nya. Sekarang bagaimana cara menampilkan isi dari halaman tertentu? Ini dia script modifikasinya

### view.php

```
<?php
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// membaca id artikel dari link
$id = $_GET['id'];

// query untuk membaca data artikel berdasar id
$query = "SELECT * FROM post WHERE id = '$id'";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

// memecah artikel berdasarkan tanda pemisah
$pecah = explode("<!--next-->", $data['konten']);

// menampilkan title
echo "<h1>".$data['title']."</h1>";

// jika parameter 'page' belum digunakan, maka akan menampilkan isi halaman pertama (array ke-0)
// jika parameter 'page' sudah diset, maka akan menampilkan isi halaman ke-i yang dituju (array ke-(i-1))
if (isset($_GET['page']))
{
    echo "<p>".$pecah[$_GET['page']-1]."</p>";
}
else echo "<p>".$pecah[0]."</p>";

// menampilkan link paging
echo "<p>Halaman: ";
for ($i=1; $i<=count($pecah); $i++)
{
    echo "<a href='view.php?id=".$data['id']."'>".$i."</a> ";
}
echo "<p>";
```

?>

OK.. sudah sedikit lebih baik sekarang scriptnya. Namun... masih ada kelemahan sedikit, yaitu bila dalam artikel tersebut tidak ada paging (mungkin karena artikelnya pendek), maka link paging halaman tersebut masih tetap muncul. So... kita mesti modifikasi sedikit dalam view.php nya menjadi:

### view.php

```
<?php
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");
```

```

// membaca id artikel dari link
$id = $_GET['id'];

// query untuk membaca data artikel berdasar id
$query = "SELECT * FROM post WHERE id = '$id'";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

// memecah artikel berdasarkan tanda pemisah
$pecah = explode("<!--next-->", $data['konten']);

// menampilkan title
echo "<h1>".$data['title']."</h1>";

// jika parameter 'page' belum digunakan, maka akan menampilkan isi halaman pertama (array ke-0)
// jika parameter 'page' sudah diset, maka akan menampilkan isi halaman ke-i yang dituju (array ke-(i-1))
if (isset($_GET['page']))
{
    echo "<p>".$pecah[$_GET['page']-1]."</p>";
}
else echo "<p>".$pecah[0]."</p>";

if (count($pecah) > 1)
{
    // menampilkan link paging jika jumlah pecahan artikelnnya > 1
    echo "<p>Halaman: ";
    for ($i=1; $i<=count($pecah); $i++)
    {
        echo "<a href='view.php?id=".$data['id']."'>".$i."</a> ";
    }
    echo "<p>";
}

?>

```

Hasil script view.php di atas tampak pada gambar di bawah ini:

## Judul 1

ini adalah halaman ketiga. ini adalah halaman ketiga.

Halaman: 1 2 3

http://localhost/view.php?id=1&page=3

Mudah bukan membuatnya? Selamat mencoba ya...

## 45. Membuat Advanced Paging dengan PHP

Paging merupakan teknik menampilkan data dengan cara membaginya ke beberapa halaman. Teknik ini diberikan untuk mengurangi scrolling window apabila data yang disajikan terlalu banyak, sehingga akan menimbulkan kejemuhan orang yang melihat dan juga akan menghasilkan page load time yang besar karena ukuran filenya besar (apabila data disajikan dalam satu halaman saja).

Untuk keperluan navigasi data biasanya disajikan dengan link berupa nomor halaman atau kadang berbentuk Previous dan Next.

Pada tutorial ini akan dipaparkan bagaimana ide membuat advance paging ini dilakukan. Advance paging adalah paging yang navigasinya berbentuk

1 2 3 4 ... 20 Next >>

Contoh tampilan di atas diasumsikan terdapat 20 halaman paging dan halaman yang sedang aktif adalah halaman 1. Bila yang aktif halaman 10, maka tampilan navigasinya adalah:

<< Prev 1 ... 7 8 9 **10** 11 12 13 ... 20 Next >>

Sedangkan bila yang aktif adalah halaman terakhir (20) maka tampilan navigasinya adalah

<< Prev 1 .. 17 18 19 **20**

Dalam contoh ini kasus advance paging ini akan diterapkan pada data guestbook atau buku tamu. Istilah ‘advance paging’ ini adalah istilah saya saja sih 😊

Pertama kita siapkan terlebih dahulu tabel untuk menyimpan datanya:

```
CREATE TABLE guestbook (
    id int(11) AUTO_INCREMENT,
    nama varchar(20),
    email varchar(20),
    tanggal date,
    komentar text,
    PRIMARY KEY (id)
)
```

Selanjutnya kita buat skenario dari paging ini. Misalkan kita akan membuat paging dengan jumlah data yang ditampilkan per halaman adalah sejumlah 5 buah. Sehingga kita perlu membuat variabel untuk menyimpan jumlah data yang akan ditampilkan per halaman.

## show.php

```
<?php  
// koneksi ke mysql  
  
mysql_connect('namahost', 'dbuser', 'dbpassword');  
mysql_select_db(data);  
  
// jumlah data yang akan ditampilkan per halaman  
  
$dataPerPage = 5;  
. . .  
?>
```

Kita lanjutkan skenarionya. Untuk script paging ini, bila dibuka pertama kali dengan URL `http://namaserver/show.php` maka otomatis menuju ke halaman 1. Untuk menuju ke halaman 2, maka URL nya `http://namaserver/show.php?page=2`. Sedangkan untuk menuju ke halaman 3, URL nya `http://namaserver/show.php?page=3`, dan seterusnya.

Dari skenario di atas tampak bahwa bila parameter `?page=...` belum diberikan pada URL maka secara otomatis menuju ke nomor halaman 1. Sedangkan bila terdapat parameter `?page=...` maka nomor halamannya menyesuaikan nilai pada parameter `?page` tersebut. Untuk mewujudkan hal ini, maka kita tambahkan perintah

```
<?php  
mysql_connect('namahost', 'dbuser', 'dbpassword');  
mysql_select_db('dbname');  
  
// jumlah data yang akan ditampilkan per halaman  
  
$dataPerPage = 5;  
  
// apabila $_GET['page'] sudah didefinisikan, gunakan nomor halaman tersebut,  
// sedangkan apabila belum, nomor halamannya 1.  
  
if(isset($_GET['page']))  
{  
    $noPage = $_GET['page'];  
}  
else $noPage = 1;  
. . .  
?>
```

Selanjutnya, dalam teknik paging perlu adanya pengaturan posisi awal data atau lebih dikenal dengan istilah offset. Offset digunakan untuk mengatur posisi awal data yang akan ditampilkan perhalaman. Perlu Anda tahu, bahwa data dalam database memiliki urutan seperti halnya array. Data pada baris pertama hasil query SQL disebut data pada posisi ke-0, data pada baris kedua disebut data pada posisi ke-1, dst.

Trus.. apa kaitannya offset dengan paging? Sabar... perhatikan dulu contoh berikut ini

Sebagai contoh, misalkan terdapat 15 data dalam tabel guestbook dimana kita ingin menyajikan pada setiap halaman adalah 5 data. Dengan demikian total terdapat 3 halaman dalam pagingnya. Untuk halaman pertama, pastilah data yang ditampilkan adalah data ke **0, 1, 2, 3** dan **4**. Sedangkan halaman kedua, data yang ditampilkan adalah **5, 6, 7, 8, dan 9**. Serta halaman ketiga, data ke **10, 11, 12, 13, 14**.

Nah... perhatikan bahwa untuk halaman 1, data awal yang ditampilkan adalah data pada urutan ke-0, halaman 2 data awalnya adalah data pada urutan ke-5 dan halaman 3 data pada urutan ke 10.

Lantas apa gunanya offset tersebut? Offset nantinya akan digunakan pada query SQL untuk memunculkan data pada setiap halaman. Query SQL yang digunakan adalah

```
SELECT * FROM guestbook LIMIT offset, jumlahData;
```

Sehingga untuk memunculkan data guestbook pada halaman pertama, maka perintahnya

```
SELECT * FROM guestbook LIMIT 0, 5;
```

Untuk memunculkan data pada halaman kedua, perintahnya:

```
SELECT * FROM guestbook LIMIT 5, 5;
```

dan untuk memunculkan data pada halaman ketiga, perintahnya:

```
SELECT * FROM guestbook LIMIT 10, 5;
```

dan seterusnya.

OK.. deh.. dah paham, trus yang jadi masalah adalah bagaimana menentukan offset pada setiap halamannya? Bila kita lihat patternnya, maka hubungan antara offset, jumlah data yang ingin ditampilkan per halaman dan nomor halamannya adalah:

**offset = (no halaman - 1) \* jumlah data per halaman;**

Lho kok bisa? he.. 3x kita cek saja. Untuk halaman 1, maka nilai offset =  $(1 - 1) * 5 = 0$ . Untuk halaman 2, nilai \$offset =  $(2 - 1) * 5 = 5$  dan halaman ketiga \$offset =  $(3 - 1) * 5 = 10$ . Bener kan? 😊

OK.. deh, dah paham tentang offset, so.. kita bisa tambahkan scriptnya untuk mencari offset dan query SQL nya.

```

<?php
mysql_connect('namahost', 'dbuser', 'dbpassword');
mysql_select_db('dbname');

// jumlah data yang akan ditampilkan per halaman

$dataPerPage = 5;

// apabila $_GET['page'] sudah didefinisikan, gunakan nomor halaman tersebut,
// sedangkan apabila belum, nomor halamannya 1.

if(isset($_GET['page']))
{
    $noPage = $_GET['page'];
}
else $noPage = 1;

// perhitungan offset

$offset = ($noPage - 1) * $dataPerPage;

// query SQL untuk menampilkan data perhalaman sesuai offset

$query = "SELECT * FROM guestbook LIMIT $offset, $dataPerPage";

$result = mysql_query($query) or die('Error');

// menampilkan data

echo "<table border='1'>";
echo "<tr><td>Nama</td><td>Email</td><td>Tanggal</td><td>Komentar</td></tr>";
while($data = mysql_fetch_array($result))
{
    echo
"<tr><td>".$data['nama']. "</td><td>".$data['email']. "</td><td>".$data['tanggal']. "</td><td>".$data['komentar']. "</td></tr>";
}

echo "</table>";

.

.

?>

```

Nah... masalah berikutnya adalah bagaimana menentukan total jumlah halamannya? Hal ini penting karena nantinya akan ditampilkan sebagai link navigasi pagingnya. Untuk menghitung total jumlah halaman, kita cari patternnya terlebih dahulu. Misalkan terdapat 15 data, dan kita ingin menyajikan data sejumlah 5 per halaman, maka total ada berapa halaman yang dibutuhkan? OK.. benar ada 3. Trus... kalo ada 31 data dan kita ingin menyajikan data sejumlah 5 per halaman, total ada berapa halaman? OK.. benar ada 7 (untuk halaman ke-7 hanya tampil 1 buah data). Sehingga dari pattern tersebut formula untuk menghitung jumlah halaman adalah

### **jumlah halaman = ceil(jumlah data / data per halaman);**

ceil() adalah function yang digunakan untuk membulatkan ke atas suatu bilangan. Misal ceil(3.2) = 4, ceil(3.7) = 4.

Trus... yang jadi masalah adalah bagaimana mendapatkan jumlah datanya? Nah... untuk mendapatkan jumlah data keseluruhan yang ada kita gunakan query SQL.

```
SELECT COUNT (*) FROM guestbook;
```

OK... I know, dan sekarang kita bisa tambahkan proses menghitung jumlah data dan jumlah halaman pada scriptnya.

```
<?php
mysql_connect('namahost', 'dbuser', 'dbpassword');
mysql_select_db('dbname');

// jumlah data yang akan ditampilkan per halaman

$dataPerPage = 5;

// apabila $_GET['page'] sudah didefinisikan, gunakan nomor halaman tersebut,
// sedangkan apabila belum, nomor halamannya 1.

if(isset($_GET['page']))
{
    $noPage = $_GET['page'];
}
else $noPage = 1;

// perhitungan offset

$offset = ($noPage - 1) * $dataPerPage;

// query SQL untuk menampilkan data perhalaman sesuai offset

$query = "SELECT * FROM guestbook LIMIT $offset, $dataPerPage";

$result = mysql_query($query);

// menampilkan data

echo "<table border='1'>";
echo "<tr><td>Nama</td><td>Email</td><td>Tanggal</td><td>Komentar</td></tr>";
while($data = mysql_fetch_array($result))
{
    echo
"<tr><td>".$data['nama']. "</td><td>".$data['email']. "</td><td>".$data['tanggal']. "</td><td>".$data['komentar']. "</td></tr>";
}

echo "</table>";

// mencari jumlah semua data dalam tabel guestbook
```

```

$query  = "SELECT COUNT(*) AS jumData FROM guestbook";
$hasil  = mysql_query($query);
$data   = mysql_fetch_array($hasil);

$jumData = $data['jumData'];

// menentukan jumlah halaman yang muncul berdasarkan jumlah semua data

$jumPage = ceil($jumData/$dataPerPage);
.

.

?>

```

Nah... sekarang tinggal bagaimana cara memunculkan link berisi nomor halaman yang menuju ke setiap halamannya. Untuk memunculkan nomor halamannya, caranya mudah. Kita hanya menggunakan looping saja.

Tapi eit.. tunggu dulu pada desain advance paging di atas, sebelum memunculkan link nomor halaman, terdapat link << prev. Kapan link ini akan muncul? ya... tepat sekali yaitu ketika nomor halaman yang sedang aktif (sedang dibuka) setelah halaman pertama atau  $\$noPage > 1$ . Trus.. menuju ke nomor halaman berapakah link tersebut? yap.. benar yaitu menuju ke nomor halaman sebelumnya ( $\$noPage - 1$ ).

Dengan demikian kita bisa tambahkan perintah berikut ini pada script

```
if ($noPage > 1) echo "<a href='". $_SERVER['PHP_SELF'] ."?page=". ($noPage-1)." '>&lt;&lt; Prev</a>";
```

sehingga menjadi

```

<?php
mysql_connect('namahost', 'dbuser', 'dbpassword');
mysql_select_db('dbname');

// jumlah data yang akan ditampilkan per halaman

$dataPerPage = 5;

// apabila $_GET['page'] sudah didefinisikan, gunakan nomor halaman tersebut,
// sedangkan apabila belum, nomor halamannya 1.

if(isset($_GET['page']))
{
    $noPage = $_GET['page'];
}
else $noPage = 1;

// perhitungan offset

$offset = ($noPage - 1) * $dataPerPage;

```

```

// query SQL untuk menampilkan data perhalaman sesuai offset

$query = "SELECT * FROM guestbook LIMIT $offset, $dataPerPage";

$result = mysql_query($query) or die('Error');

// menampilkan data

echo "<table border='1'>";
echo "<tr><td>Nama</td><td>Email</td><td>Tanggal</td><td>Komentar</td></tr>";
while($data = mysql_fetch_array($result))
{
    echo
"<tr><td>".$data['nama']. "</td><td>".$data['email']. "</td><td>".$data['tanggal']. "</td><td>".$data['komentar']. "</td></tr>";
}

echo "</table>";

// mencari jumlah semua data dalam tabel guestbook

$query = "SELECT COUNT(*) AS jumData FROM guestbook";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

$jumData = $data['jumData'];

// menentukan jumlah halaman yang muncul berdasarkan jumlah semua data

$jumPage = ceil($jumData/$dataPerPage);

// menampilkan link previous

if ($noPage > 1) echo "<a href='". $_SERVER['PHP_SELF'] . "?page=". ($noPage-1). "'>&lt;&lt; Prev</a>";
.
.
?>

```

Berikutnya kita akan tampilkan link nomor-nomor halamannya menggunakan looping. Adapun perintah-perintahnya adalah seperti di bawah ini.

```

for($page = 1; $page <= $jumPage; $page++)
{
    if (((($page >= $noPage - 3) && ($page <= $noPage + 3)) || ($page == 1) || ($page == $jumPage)))
    {
        if (($showPage == 1) && ($page != 2)) echo "...";
        if (($showPage != ($jumPage - 1)) && ($page == $jumPage)) echo "...";
        if ($page == $noPage) echo " <b>". $page. "</b> ";
        else echo " <a
href='". $_SERVER['PHP_SELF'] . "?page=". $page. "'>". $page. "</a> ";
        $showPage = $page;
    }
}

```

```
}
```

Konsep loopingnya sederhana, yaitu menuliskan link nomor halaman mulai dari 1 s/d jumlah halamannya. Namun harap dicatat di sini bahwa karena tidak semua nomor halaman ditampilkan sesuai desain sebelumnya maka kita perlu berikan syarat.

Maksud dari syarat

```
if (((($page >= $noPage - 3) && ($page <= $noPage + 3)) || ($page == 1) ||  
($page == $jumPage))  
{  
    .  
    .  
}
```

adalah hanya memunculkan link nomor halaman 1 dan juga halaman terakhir, sekaligus pula nomor-nomor halaman dalam rentang 3 halaman sebelum dan sesudah nomor halaman yang aktif. Contoh jika halaman yang aktif adalah halaman 7 dari 15 halaman, maka nanti diharapkan nomor halaman yang muncul adalah

1 4 5 6 **7** 8 9 10 15

Anda dapat mengubah rentang halaman sesuai yang diinginkan.

Perintah

```
if (($showPage == 1) && ($page != 2)) echo "...";
```

Digunakan untuk memunculkan tanda ‘...’ di antara halaman nomor 1 dengan halaman berikutnya yang tampil. Tapi perlu diingat bahwa tanda ini akan muncul bila nomor halaman yang muncul setelah 1 ialah bukan 2. Apabila setelah 1 ini muncul 2, maka tanda ini tidak muncul.

Demikian pula untuk perintah

```
if (($showPage != ($jumPage - 1)) && ($page == $jumPage)) echo "...";
```

Perintah di atas digunakan untuk memunculkan tanda ‘...’ sebelum nomor halaman terakhir bila halaman sebelumnya yang muncul bukan ‘nomor halaman terakhir – 1’.

Sehingga dari kedua perintah di atas diharapkan akan menampilkan nomor halaman seperti di bawah ini

1 ... 4 5 6 **7** 8 9 10 ... 15 (bila yang aktif adalah halaman 7)

1 2 **3** 4 5 6... 15 (bila yang aktif adalah halaman 2)

1 ... 10 11 12 **13** 14 15 (bila yang aktif adalah halaman 13)

Sedangkan perintah

```
if ($page == $noPage) echo "<b>". $page . "</b> ";
else echo "<a href='". $_SERVER['PHP_SELF'] ."?page=". $page . "'>". $page . "</a> ";
```

digunakan untuk menampilkan link nomor halaman. Namun link dari nomor halaman ini ditampilkan hanya pada nomor halaman yang sedang tidak aktif. Sedangkan pada nomor halaman yang sedang aktif nomor halaman hanya dicetak tebal saja untuk menunjukkan bahwa halaman yang sedang aktif adalah halaman tersebut.

Terakhir.. kita perlu link untuk memunculkan next >>. OK kapan link ini muncul? ya benar... yaitu ketika halaman yang sedang aktif bukanlah pada halaman terakhir atau sebelum halaman terakhir. Sehingga kita bisa tambahkan perintah

```
if ($noPage < $jumPage) echo "<a href='". $_SERVER['PHP_SELF'] ."?page=". ($noPage+1) . "'>Next &gt;&gt;". </a> ";
```

pada script, sehingga scriptnya menjadi

```
<?php
mysql_connect('namahost', 'dbuser', 'dbpassword');
mysql_select_db('dbname');

// jumlah data yang akan ditampilkan per halaman
$dataPerPage = 5;

// apabila $_GET['page'] sudah didefinisikan, gunakan nomor halaman tersebut,
// sedangkan apabila belum, nomor halamannya 1.

if(isset($_GET['page']))
{
    $noPage = $_GET['page'];
}
else $noPage = 1;

// perhitungan offset

$offset = ($noPage - 1) * $dataPerPage;

// query SQL untuk menampilkan data perhalaman sesuai offset

$query = "SELECT * FROM guestbook LIMIT $offset, $dataPerPage";

$result = mysql_query($query) or die('Error');

// menampilkan data

echo "<table border='1'>";
echo "<tr><td>Nama</td><td>Email</td><td>Tanggal</td><td>Komentar</td></tr>";
while($data = mysql_fetch_array($result))
{

```

```

echo
"<tr><td>".$data['nama']. "</td><td>".$data['email']. "</td><td>".$data['tangga
l']. "</td><td>".$data['komentar']. "</td></tr>";
}

echo "</table>";

// mencari jumlah semua data dalam tabel guestbook

$query = "SELECT COUNT(*) AS jumData FROM guestbook";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

$jumData = $data['jumData'];

// menentukan jumlah halaman yang muncul berdasarkan jumlah semua data

$jumPage = ceil($jumData/$dataPerPage);

// menampilkan link previous

if ($noPage > 1) echo "<a href='". $_SERVER['PHP_SELF'] ."?page=". ($noPage-
1).">&lt;&lt; Prev</a>";

// memunculkan nomor halaman dan linknya

for($page = 1; $page <= $jumPage; $page++)
{
    if (((($page >= $noPage - 3) && ($page <= $noPage + 3)) || ($page ==
1) || ($page == $jumPage))
    {
        if (($showPage == 1) && ($page != 2)) echo "...";
        if (($showPage != ($jumPage - 1)) && ($page == $jumPage)) echo
"...";
        if ($page == $noPage) echo " <b>".$page."</b> ";
        else echo " <a
href='". $_SERVER['PHP_SELF'] ."?page=".$page."'>".$page."</a> ";
        $showPage = $page;
    }
}

// menampilkan link next

if ($noPage < $jumPage) echo "<a
href='". $_SERVER['PHP_SELF'] ."?page=". ($noPage+1) ."'>Next &gt;&gt;</a>";

?>

```

Wah panjang juga ya artikelnya.. tapi gak papa deh, mudah-mudahan ada manfaatnya bagi Anda. Selamat mencoba dan berpaginaing ria 😊

## 46. Input Data Ke MySQL Secara Simultan Via Form dengan PHP

Seringkali kita membuat script untuk input data hanya terdiri dari sebuah data atau record saja. Itu sangat-sangat mudah dilakukan. Lantas, bagaimana bila sekali entri langsung beberapa record sekaligus secara simultan? Dengan demikian kita bisa entri beberapa data sekaligus tanpa repot bolak-balik membuka form baru ketika mau menyisipkan data. Kira-kira hal ini bisa dilakukan nggak ya? He.. 3x... Ya jelas bisa donk! Oleh karena itu akan saya paparkan bagaimana ide membuat script seperti itu dengan PHP melalui artikel ini.

Sebelum kita mulai membahas cara pembuatannya, ada baiknya Anda simak gambar berikut ini yang merupakan grand design studi kasus yang akan kita buat sebagai contoh input data simultan ini. Studi kasus yang saya ambil terkait dengan entri data mahasiswa.

Pertama, user diminta memasukkan jumlah data yang mau dientri terlebih dahulu (lihat gambar di bawah ini)

### Input Data Mahasiswa

Jumlah Data

Kedua, setelah user memasukkan jumlah data yang mau dientri, selanjutnya muncul form dengan komponen entri sejumlah data yang tadi dimasukkan. Pada form inilah data yang mau disimpan ke database, dimasukkan.

### Input Data Mahasiswa

NIM	NAMA MHS
M001	AAA
M002	BBB
M003	CCC
M004	DDD
M005	EEE

Setelah form di atas disubmit, maka data akan tersimpan ke database, seperti tampak pada gambar berikut ini

nim	namamhs
M001	AAA
M002	BBB
M003	CCC
M004	DDD
M005	EEE

Nah... bagaimana cara membuatnya?

Pertama kita siapkan dahulu tabel database MySQL nya. Pada studi kasus ini, kita buat simpel saja deh (cukup 2 field saja dulu)

```
CREATE TABLE mhs (
    nim varchar(5),
    namamhs varchar(20),
    PRIMARY KEY (nim)
);
```

Kemudian kita buat form untuk memasukkan jumlah datanya terlebih dahulu.

### form.html

```
<h1>Input Data Mahasiswa</h1>

<form method="post" action="submit.php">
Jumlah Data <input type="text" name="jum"> <input type="submit" name="submit"
value="Submit">
</form>
```

Saya kira untuk membuat form di atas tidak ada masalah kan? 😊

Berikutnya kita buat script untuk menampilkan komponen inputnya sejumlah data yang tadi dimasukkan pada form di atas. Konsepnya adalah kita gunakan looping untuk menampilkan komponen inputnya. Lantas... loopingnya dilakukan berapa kali? ya... sebanyak jumlah datanya donk.

### submit.php

```
<h1>Input Data Mahasiswa</h1>

<form method="post" action="insert.php">
<table border="1">
<tr><td>NIM</td><td>NAMA MHS</td></tr>

<?php
$n = $_POST['jum']; // membaca jumlah data

for ($i=1; $i<=$n; $i++)
{
    echo "<tr><td><input type='text' name='nim".$i."'></td><td><input
type='text' name='nama".$i."'></td></tr>";
```

```

}

?>

</table>
<br>
<input type="hidden" name="jum" value="<?php echo $n; ?>">
<input type="submit" name="submit" value="Submit">
</form>

```

Bila Anda perhatikan script di atas, maka nama komponen untuk input NIM berbentuk

```
<input type='text' name='nim".$i."'>
```

yang mana nantinya looping ( $\$i=1, 2, 3, \dots, \$n$ ) akan membentuk komponen-komponen dengan nama

```

<input type='text' name='nim1'>
<input type='text' name='nim2'>
<input type='text' name='nim3'>
.
.
dst

```

Nah... hal yang sama kita buat juga untuk komponen input nama mahasiswa.

Kemudian, coba perhatikan pula bahwa di bagian akhir script terdapat perintah

```
<input type="hidden" name="jum" value="<?php echo $n; ?>">
```

Perintah tersebut akan menghasilkan komponen hidden untuk menyimpan nilai  $\$n$  (banyaknya data). Nilai ini nanti akan ikut disubmit ke script berikutnya guna pemrosesan lebih lanjut.

Terakhir.. kita buat script untuk insert datanya. Untuk membuat script insert data secara simultan ini, mungkin yang menjadi kendala adalah bagaimana cara membaca semua data yang telah diinputkan dalam form sebelumnya untuk kemudian disimpan ke dalam database. Adapun konsep pembacaan datanya adalah dengan menggunakan looping juga. Looping ini dilakukan sebanyak jumlah datanya ( $\$n$ ). Oleh karena itulah pada script ini kita butuh nilai  $\$n$  yang ikut tersubmit dari script sebelumnya.

## **insert.php**

```

<?php

// koneksi ke mysql
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

$n = $_POST['jum']; // membaca jumlah data

// looping

```

```

for ($i=1; $i<=$n; $i++)
{
    $datanim = $_POST['nim' . $i];
    $datanama = $_POST['nama' . $i];

    if ((!empty($datanim)) && (!empty($datanama)))
    {
        $query = "INSERT INTO mhs (nim, namamhs) VALUES ('$datanim',
'$datanama')";
        $hasil = mysql_query($query);

        if ($hasil) echo "Input data sukses<br>";
        else echo "Input data gagal<br>";
    }
}

?>

```

Perhatikan perintah `$datanim = $_POST['nim' . $i];` yang terdapat dalam looping di atas. Perintah tersebut digunakan untuk membaca value yang diinputkan dari komponen form bernama ‘nim1’, ‘nim2’, ... dst. Hal yang sama juga berlaku untuk perintah `$datanama = $_POST['nama' . $i];`. Setelah value NIM dan Nama Mhs dari komponen pertama (‘nim1’ dan ‘nama1’) dibaca, selanjutnya baru disimpan ke database. Kemudian pembacaan dan insert data ke database dilanjutkan pada komponen kedua (‘nim2’ dan ‘nama2’), begitu seterusnya sampai dengan komponen terakhir.

Oya.. tambahan lagi, dalam script di atas juga terdapat statement

```

if ((!empty($datanim)) && (!empty($datanama)))
{
    ...
}

```

Statement tersebut digunakan untuk mengecek jika ada baris komponen form yang tidak diisi NIM maupun nama mahasiswanya. Proses insert data ke database hanya akan dilakukan bila komponen input NIM dan nama mahasiswanya tidak kosong.

OK deh... demikian tadi ide untuk membuat script insert data ke database MySQL secara simultan melalui form dengan script PHP<sup>[1]</sup>. Mudah-mudahan ada manfaatnya.

## 47. Script PHP Updating Data Secara Simultan (Studi Kasus Update Nilai Kuliah Mahasiswa)

Ada seorang member saya <sup>[3]</sup>, mas Christ, menanyakan sebuah kasus. Dalam kasusnya diceritakan bahwa ia ingin membuat aplikasi untuk keperluan update nilai siswa. Namun dia tidak ingin proses update nilainya satu-persatu, namun bisa secara simultan sekaligus. Proses ini terjadi ketika dalam proses entri nilai mata pelajaran.

Perlu diketahui bahwa ketika siswa mengambil pelajaran, terlebih dahulu data siswa dan mata pelajaran yang diikutinya disimpan ke dalam database, namun belum ada nilainya (nilainya 0). Begitu nilai sudah ada dan ingin diinputkan, maka proses updatelah yang terjadi. Nah... saudara Christ ini menghendaki proses updatenya bisa dilakukan sekaligus secara simultan dan tidak satu-satu. Kalau satu-satu jadi repot, apalagi kalau jumlah siswanya banyak.

So... dalam artikel ini akan dibahas teknik atau ide untuk membuat aplikasi seperti yang dimaksud oleh member saya tersebut, namun dengan sedikit modifikasi dalam kasusnya. Kali ini saya ambil kasus proses update nilai kuliah mahasiswa. Ya... supaya lebih mudah saja saya memahaminya, karena saya orang kampus kalee 😊Nggak... kok just kidding, saya pilih kasus ini karena dalam komputer saya sudah ada tabel dan data terkait dengan hal ini. Jadi gak perlu repot-repot bikin tabel dan data baru. Tapi yang jelas konsepnya sama seperti yang diinginkan mas Christ kok.

Untuk keperluan pembahasannya dalam studi kasus ini, kita siapkan terlebih dahulu tabel dan datanya

```
CREATE TABLE mhs (
    nim varchar(10),
    namamhs varchar(30),
    alamat text,
    sex varchar(10),
    PRIMARY KEY (nim)
);
CREATE TABLE mk (
    kodemk varchar(5),
    namamk varchar(20),
    sks int(11),
    smt int(11),
    PRIMARY KEY (kodemk)
);
CREATE TABLE ambilmk (
    nim varchar(10),
    kodemk varchar(5),
    nilai float,
    PRIMARY KEY (nim, kodemk)
);
INSERT INTO mhs VALUES ('M0197001', 'ROSIHAN ARI YUANA', 'COLOMADU', 'L');
INSERT INTO mhs VALUES ('M0197002', 'DWI AMALIA FITRIANI', 'KUDUS', 'P');
INSERT INTO mhs VALUES ('M0197003', 'FAZA FAUZAN KH.', 'COLOMADU', 'L');
```

```

INSERT INTO mhs VALUES ('M0197004', 'NADA HASANAH', 'COLOMADU', 'P');
INSERT INTO mhs VALUES ('M0197005', 'MUH. AHSANI TAQWIM', 'COLOMADU', 'L');
INSERT INTO mk VALUES ('K001', 'KALKULUS II', 3, 4);
INSERT INTO mk VALUES ('K004', 'NUMERIK', 3, 4);
INSERT INTO ambilmk VALUES ('M0197001', 'K001', 0);
INSERT INTO ambilmk VALUES ('M0197002', 'K001', 0);
INSERT INTO ambilmk VALUES ('M0197003', 'K001', 0);
INSERT INTO ambilmk VALUES ('M0197004', 'K001', 0);
INSERT INTO ambilmk VALUES ('M0197005', 'K004', 0);

```

Perhatikan pada sampel tabel dan data di atas bahwa, misalkan hanya terdapat 2 matakuliah yang ada (Kalkulus II dan Numerik). Sedangkan mahasiswa yang mengambil ‘Kalkulus II’ ada 4 orang yaitu yang bernim ‘M0197001’, ‘M0197002’, ‘M0197003’, dan ‘M0197004’. Tampak pula bahwa nilai matakuliah tersebut untuk keempat mahasiswa adalah masih 0 (belum diinputkan nilainya).

Selanjutnya kita coba desain skenario untuk proses update nilai secara simultan ini. Skenario pertama adalah memilih nama matakuliah yang akan kita update nilainya (lihat gambar di bawah ini)

Pilih Mata Kuliah :

KALKULUS II
NUMERIK

Setelah dipilih nama matakuliah, akan muncul daftar form nilai untuk semua mahasiswa yang mengambil matakuliah tersebut. Nah.. di form inilah kita kan lakukan proses entri nilai atau edit nilai untuk mahasiswanya.

## Update Nilai

No	NIM	Nilai
1	M0197001	0
2	M0197002	0
3	M0197003	0
4	M0197004	0

## Update Nilai

No	NIM	Nilai
1	M0197001	70
2	M0197002	80
3	M0197003	70
4	M0197004	85



Contoh tampilan di atas menggambarkan proses update nilai mahasiswa yang mengambil ‘Kalkulus II’

OK.. skenario sudah, berikutnya kita mulai buat aplikasinya.

Pertama kita buat form untuk memilih nama matakuliahnya terlebih dahulu.

### formnilai.php

```
<?php
// koneksi ke mysql
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");
?>

<form method="post" action="submit.php">
Pilih Mata Kuliah :
<select name="mk">
<?php
// query untuk menampilkan semua matakuliah dari tabel 'mk'
$query = "SELECT * FROM mk";
$hasil = mysql_query($query);
while ($data = mysql_fetch_array($hasil))
{
    echo "<option value='".$data['kodemk']."'>".$data['namamk']. "</option>";
}
?>
<input type="submit" value="Submit" name="submit" />
</select>
</form>
```

### Keterangan:

Nilai yang disubmit oleh form di atas menuju submit.php adalah kode matakuliahnya.

Selanjutnya kita buat script submit.php. Script ini nanti akan memunculkan form update nilai untuk semua mahasiswa yang mengambil matakuliah yang dipilih dari formnilai.php

### submit.php

```
<h1>Update Nilai</h1>

<form method="post" action="update.php">
<table border="1">
<tr><td>No</td><td>NIM</td><td>Nilai</td></tr>

<?php
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// membaca kode matakuliah yang disubmit dari formnilai.php
$kodeMK = $_POST['mk'];

// menampilkan data nim dan nilai mahasiswa yang mengambil matakuliah
// berdasarkan kode MK
$query = "SELECT nim, nilai FROM ambilmk WHERE kodemk = '$kodeMK'";

$hasil = mysql_query($query);

// inisialisasi counter
$i = 1;
while ($data = mysql_fetch_array($hasil))
{
    echo "<tr><td>".$i."</td><td>".$data['nim']."</td><td><input type='hidden' name='mhs' value='".$data['nim']."' />
          <input type='text' name='nilai' value='".$data['nilai']."' />
        </td></tr>";
    $i++;
}
$jumMhs = $i-1;
?>
</table><br />
<input type="hidden" name="n" value="<?php echo $jumMhs ?>" />
<input type="submit" value="Update" name="submit" />
</form>
```

### Keterangan:

Perintah `<input type='hidden' name='mhs' value='".$data['nim']."' />` digunakan untuk membuat komponen hidden yang berisi nim mahasiswa yang mengambil matakuliah tersebut. NIM ini nantinya akan ikut disubmit ketika proses updating nilai. Mengapa NIM perlu disubmit ketika proses updating nilai? ya... karena NIM akan diperlukan dalam query SQL update nilai yang berbentuk

`UPDATE ambilmk SET nilai = ... WHERE nim = ...`

Query SQL di atas nanti akan diletakkan dalam script update.php

Dalam script di atas terdapat pula perintah <input type="hidden" name="n" value="<?php echo \$jumlahMhs ?>" />. Nilai 'n' ini nanti akan ikut disubmit untuk proses looping dalam membaca semua nilai dan NIM yang berasal dari form sekaligus menjalankan query SQL update data. Proses pembacaan dan menjalankan query update data ini dilakukan sebanyak n kali (sebanyak jumlah mahasiswa yang mengambil).

Terakhir... kita buat script update.php nya

### update.php

```
<?php
mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// membaca jumlah mahasiswa (n) dari submit.php
$jumlahMhs = $_POST['n'];

// proses looping untuk membaca nilai dan nim mahasiswa dari form, serta
// menjalankan query update
for ($i=1; $i<=$n; $i++)
{
    // membaca nim mahasiswa ke-i, i = 1, 2, 3, ..., n
    $nimMhs = $_POST['mhs'].$i;

    // membaca nilai mahasiswa ke-i, i = 1, 2, 3, ..., n
    $nilai = $_POST['nilai'].$i;

    // update nilai mahasiswa ke-i, i = 1, 2, 3, ..., n
    $query = "UPDATE ambilmk SET nilai = '$nilai' WHERE nim = '$nimMhs'";
    mysql_query($query);
}

echo "<h2>Update nilai sukses</h2>";
?>
```

Nah... mudah dan simpel bukan membuatnya?

Dalam kasus yang nyata, tentu struktur tabel dan datanya tidak sesimpel yang saya bahas di atas namun kemungkinan jauh lebih kompleks. Di sini saya hanya memberikan ide dan konsep dasarnya saja untuk proses update data secara simultan. Untuk selanjutnya silakan Anda kembangkan sendiri sesuai kasus yang Anda hadapi masing-masing. OK?

## 48. Membuat Script PHP untuk Hapus Data dengan Multi Selection

Pernahkah Anda membuat script untuk menghapus data dengan konsep multi selection? Untuk menghapus sebuah data sih.. saya kira mudah untuk membuatnya. Pembahasan tentang hal ini telah saya kupas tuntas di [artikel sebelumnya](#)<sup>[1]</sup>.

Tapi... bagaimana bila data yang akan dihapus lebih dari satu sekaligus? Rumit gak ya membuat scriptnya? Hmm... nggak rumit kok, cuman sedikit memusingkan saja he.. he.. he 😊 nggak ding, becanda... mudah-mudahan artikel yang saya sajikan ini tidak membuat Anda pusing.

OK... konsep menghapus data dengan multi selection ini biasanya menggunakan bantuan komponen checkbox. Mengapa checkbox? ya.. karena komponen checkbox memungkinkan kita bisa memilih pilihan lebih dari satu.

Data yang akan dihapus terlebih dahulu dipilih dengan memberi tanda check pada checkbox yang ada di depan setiap data. Perhatikan contoh tampilan berikut ini:

<input type="checkbox"/> Pilih Semua	NIM	Nama Mahasiswa
<input type="checkbox"/>	M0197001	ROSIHAN ARI YUANA
<input checked="" type="checkbox"/>	M0197002	DWI AMALIA FITRIANI
<input type="checkbox"/>	M0197003	FAZA FAUZAN KH.
<input checked="" type="checkbox"/>	M0197004	NADA HASANAH
<input checked="" type="checkbox"/>	M0197005	MUH. AHSANI TAQWIM

**hapus** **Batal**

[2]

Nah... kita akan membuat script PHP untuk menghapus data dengan konsep multi selection seperti gambaran di atas.

Untuk studi kasus yang diambil, lagi-lagi kita mengambil kasus tentang data mahasiswa 😊 Huh... mentang-mentang yang buat artikel ini seorang dosen kali ya 😊

Misalkan kita punya tabel dan data sbb:

```
CREATE TABLE mhs (
    nim varchar(10),
    namamhs varchar(30),
    PRIMARY KEY (nim)
)
INSERT INTO mhs VALUES ('M0197001', 'ROSIHAN ARI YUANA');
INSERT INTO mhs VALUES ('M0197002', 'DWI AMALIA FITRIANI');
INSERT INTO mhs VALUES ('M0197003', 'FAZA FAUZAN KH.');
INSERT INTO mhs VALUES ('M0197004', 'NADA HASANAH');
INSERT INTO mhs VALUES ('M0197005', 'MUH. AHSANI TAQWIM');
```

Langkah pertama dalam membuat script penghapusan data ini, terlebih dahulu kita siapkan script untuk menampilkan semua datanya. Untuk setiap data nantinya kita letakkan sebuah checkbox di depannya yang digunakan untuk memilih data mana saja yang akan dihapus.

### **data.php**

```
<html>
<head>
    <title>Hapus data mahasiswa</title>
</head>
<body>
<h1>Hapus Data Mahasiswa</h1>
<?php

// koneksi mysql

mysql_connect ("dbhost", "dbuser", "dbpass");
mysql_select_db ("dbname");

// query SQL untuk menampilkan semua data

$query = "SELECT * FROM mhs";
$hasil = mysql_query ($query);

// membuat form penghapusan data

echo "<form name='myform' method='post'
action='".$_SERVER['PHP_SELF']."'?action=del'>";
echo "<table border='1'>";
echo "<tr><td></td><td><b>NIM</b></td><td><b>Nama Mahasiswa</b></td></tr>";
$i = 0;
while($data = mysql_fetch_array($hasil))
{
    echo "<tr><td><input type='checkbox' name='nim".$i."'"
value='".$data['nim']."'"
/></td><td>".$data['nim']."'</td><td>".$data['namamhs']."'</td></tr>";
    $i++;
}
echo "</table>";
echo "<input type='hidden' name='n' value='".$i."' />";
echo "<p><input type='submit' value='Hapus' name='submit'> <input
type='reset' value='Batal' name='reset'></p>";
echo "</form>";
?>

</body>
</html>
```

Perhatikan script di atas! Form nantinya akan diarahkan ke script itu sendiri (menggunakan \$\_SERVER['PHP\_SELF']) namun dengan parameter '?action=del'. Anda boleh arahkan pemrosesan formnya ke script lain. It's up to you.

Selain itu perhatikan pula bahwa untuk setiap komponen checkbox diberikan name yang berbeda. Di sini diberikan name berbentuk 'nim0', 'nim1', 'nim2', ... dst (perhatikan bagian name='nim".\$i."').

Kemudian, terdapat pula komponen hidden value (name='n') yang bernilai sama dengan jumlah data total atau dalam hal ini adalah \$i. Komponen ini nantinya akan sangat penting ketika memproses form terutama pada saat membaca komponen checkbox yang dipilih.

Selanjutnya untuk pemanis, misalkan kita ingin membuat fasilitas SELECT ALL untuk memilih semua data sekaligus. OK Bagaimana caranya? Ya.. kita gunakan Javascript. Oya... jangan lupa berikan sebuah checkbox tambahan untuk memberikan fasilitas ini. Begitu checkbox ini dipilih, maka semua checkbox yang ada di depan data akan terpilih. Untuk hal ini, kita berikan event onclick() pada checkbox tsb. Mmm... sebaiknya checkbox ini diletakkan di dalam header tabel saja ya.

### **data.php**

```
<html>
<head>
    <title>Hapus data mahasiswa</title>
    <script type="text/javascript">

        function pilihan()
        {
            // membaca jumlah komponen dalam form bernama 'myform'
            var jumKomponen = document.myform.length;

            // jika checkbox 'Pilih Semua' dipilih
            if (document.myform[0].checked == true)
            {
                // semua checkbox pada data akan terpilih
                for (i=1; i<=jumKomponen; i++)
                {
                    if (document.myform[i].type == "checkbox")
document.myform[i].checked = true;
                }
            }
            // jika checkbox 'Pilih Semua' tidak dipilih
            else if (document.myform[0].checked == false)
            {
                // semua checkbox pada data tidak dipilih
                for (i=1; i<=jumKomponen; i++)
                {
                    if (document.myform[i].type == "checkbox")
document.myform[i].checked = false;
                }
            }
        }

    </script>
</head>
<body>
<h1>Hapus Data Mahasiswa</h1>
```

```

<?php

// koneksi mysql

mysql_connect ("dbhost", "dbuser", "dbpass");
mysql_select_db ("dbname");

// query SQL untuk menampilkan semua data

$query = "SELECT * FROM mhs";
$hasil = mysql_query ($query);

// membuat form penghapusan data

echo "<form name='myform' method='post'
action='".$_SERVER['PHP_SELF']."' ?action=del'>";
echo "<table border='1'>";
echo "<tr><td><input type='checkbox' name='pilih' onclick='pilihan()' />
Pilih semua</td><td><b>NIM</b></td><td><b>Nama Mahasiswa</b></td></tr>";
$i = 0;
while ($data = mysql_fetch_array ($hasil))
{
    echo "<tr><td><input type='checkbox' name='nim'".$i."'
value='".$data['nim']."'"
    /></td><td>".$data['nim']."</td><td>".$data['namamhs']."</td></tr>";
    $i++;
}
echo "</table>";
echo "<input type='hidden' name='n' value='".$i."' />";
echo "<p><input type='submit' value='Hapus' name='submit'> <input
type='reset' value='Batal' name='reset'></p>";
echo "</form>";
?>

</body>
</html>

```

Nah... selanjutnya, kita bisa membuat bagian untuk memproses formnya atau membuat script untuk proses penghapusan datanya.

Trus... bagaimana konsep membaca data mana saja yang akan dihapus sesuai pilihan checkboxnya? Untuk membaca data mana yang akan dihapus, kita gunakan bantuan looping. Berapa jumlah loopingnya? Jumlah loopingnya sesuai jumlah komponen checkboxnya. Nah... di sinilah pentingnya penggunaan komponen hidden value `<input type='hidden' name='n' value="" . $i . ">"` Dalam contoh ini nilai n=5.

Selanjutnya isi dalam loopingnya bagaimana? ya... kita cek saja apakah komponen checkbox dengan name='nim0' dipilih tidak? jika ya, maka baca NIM mahasiswa terkait dengan komponen tersebut, lalu jalankan query SQL untuk menghapus data mahasiswanya. Trus... komponen checkbox dengan name='nim1' dicek apakah dipilih atau tidak. Jika ya, maka baca NIM mahasiswa terkait dengan komponen tersebut, lalu hapus dengan query SQL. Proses ini diulang sebanyak 5 kali untuk kasus di atas, yaitu mulai dari komponen checkbox dengan name='nim0' sampai dengan name='nim4'.

Lantas yang menjadi masalah adalah, bagaimana cara kita mendeteksi bahwa komponen checkbox dengan name tertentu itu dipilih atau tidak? caranya adalah kita gunakan statement IF seperti di bawah ini

```
if (isset($_POST['nim'][$i]))  
{  
    ...  
}
```

dengan \$i = 0, 1, ..., n-1.

Jika suatu komponen checkbox itu dipilih, maka perintah isset(\$\_POST['nim'][\$i]) akan menghasilkan nilai TRUE. Sedangkan FALSE jika tidak dipilih.

Dengan demikian, script untuk menghapus data dengan multi selection ini adalah sbb:

### **data.php**

```
<html>  
<head>  
    <title>Hapus data mahasiswa</title>  
    <script type="text/javascript">  
  
        function pilihan()  
        {  
            // membaca jumlah komponen dalam form bernama 'myform'  
            var jumKomponen = document.myform.length;  
  
            // jika checkbox 'Pilih Semua' dipilih  
            if (document.myform[0].checked == true)  
            {  
                // semua checkbox pada data akan terpilih  
                for (i=1; i<=jumKomponen; i++)  
                {  
                    if (document.myform[i].type == "checkbox")  
document.myform[i].checked = true;  
                }  
            }  
            // jika checkbox 'Pilih Semua' tidak dipilih  
            else if (document.myform[0].checked == false)  
            {  
                // semua checkbox pada data tidak dipilih  
                for (i=1; i<=jumKomponen; i++)  
                {  
                    if (document.myform[i].type == "checkbox")  
document.myform[i].checked = false;  
                }  
            }  
        }  
  
</script>  
</head>  
<body>  
<h1>Hapus Data Mahasiswa</h1>
```

```

<?php

// koneksi mysql

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// bagian script untuk menghapus data

if ($_GET['action'] == "del")
{
    // membaca nilai n dari hidden value
    $n = $_POST['n'];

    for ($i=0; $i<=$n-1; $i++)
    {
        if (isset($_POST['nim'][$i]))
        {
            $nim = $_POST['nim'][$i];
            $query = "DELETE FROM mhs WHERE nim = '$nim'";
            mysql_query($query);
        }
    }
}

// query SQL untuk menampilkan semua data

$query = "SELECT * FROM mhs";
$hasil = mysql_query($query);

// membuat form penghapusan data

echo "<form name='myform' method='post'
action='".$._SERVER['PHP_SELF']."?action=del'>";
echo "<table border='1'>";
echo "<tr><td><input type='checkbox' name='pilih' onclick='pilihan()' />
Pilih semua</td><td><b>NIM</b></td><td><b>Nama Mahasiswa</b></td></tr>";
$i = 0;
while($data = mysql_fetch_array($hasil))
{
    echo "<tr><td><input type='checkbox' name='nim'.".$i."'
value='".$data['nim']."' /><td>".$data['nim']. "</td><td>".$data['namamhs']. "</td></tr>";
    $i++;
}
echo "</table>";
echo "<input type='hidden' name='n' value='".$i."' />";
echo "<p><input type='submit' value='Hapus' name='submit'> <input
type='reset' value='Batal' name='reset'></p>";
echo "</form>";

?>

</body>
</html>

```

## 49. Membuat Script Grafik PHP Untuk Menampilkan Indeks Prestasi Mahasiswa (Studi Kasus)

Dalam [artikel sebelumnya](#)<sup>[1]</sup>, saya pernah singgung bagaimana membuat query SQL untuk menampilkan Indeks Prestasi (IP) mahasiswa dalam setiap semester yang telah ditempuhnya. Hasil query ini nanti bisa digunakan untuk melihat progress atau perkembangan prestasi akademik mahasiswa. Oleh karena itu biasanya, query ini sering digunakan dalam suatu sistem informasi akademik.

Selanjutnya dalam artikel ini, saya akan mencoba memaparkan bagaimana menampilkan hasil query SQL di atas ke dalam bentuk grafik (chart) menggunakan script PHP, atau dengan kata lain artikel ini akan membahas bagaimana membuat grafik (chart) untuk menampilkan perkembangan IP setiap semester mahasiswa.

Mengapa harus ditampilkan dalam bentuk grafik (chart)? ya... karena dengan grafik, seseorang dapat dengan mudah membaca suatu informasi. Dengan grafik, seseorang dapat dengan mudah mengetahui apakah prestasi akademiknya semakin turun, ataukah naik atau sedang-sedang saja.

OK deh... trus bagaimana skenario dari script ini? adapun skenarionya adalah pertama akan disajikan terlebih dahulu daftar (list) dari mahasiswa yang ada. Kemudian disetiap mahasiswa terdapat link untuk melihat grafik IP nya. Berikut ini adalah gambarannya.

### Data Mahasiswa

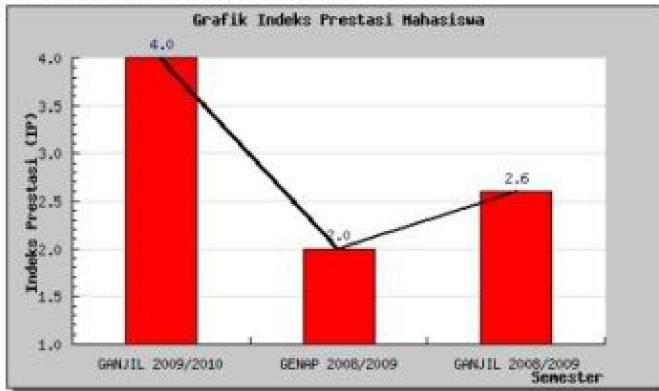
NIM	Nama Mhs	Statistik IP
M002	DWI AMALIA FITRIANI	<a href="#">View</a>
M003	FAZA FAUZAN	<a href="#">View</a>

Apabila link VIEW tersebut diklik, maka akan tampil grafik IP untuk setiap semesternya

## Statistik Indeks Prestasi

NIM : M002

Nama Mhs : DWI AMALIA FITRIANI



OK.. paham ya skenarionya? Nah.. selanjutnya kita mulai rancang tabel-tabel dan field yang diperlukan untuk databasenya. Pada prinsipnya, struktur tabelnya sama seperti di [artikel sebelumnya](#)<sup>[1]</sup>.

Terdapat 3 tabel yang kita perlukan untuk studi kasus ini, yaitu ‘mhs’ (tabel mahasiswa), ‘mk’ (tabel matakuliah) dan ‘ambilmk’ (tabel pengambilan matakuliah)

```
CREATE TABLE mhs (
    nim varchar(5),
    nama varchar(20),
    PRIMARY KEY (nim)
)
CREATE TABLE mk (
    kodeMK varchar(5),
    namaMK varchar(20),
    sks int(11),
    PRIMARY KEY (kodeMK)
)
CREATE TABLE ambilmk (
    nim varchar(5),
    kodeMK varchar(5),
    nilai float,
    smt varchar(10) DEFAULT NULL,
    tahun varchar(9) DEFAULT NULL,
    PRIMARY KEY (nim, kodeMK)
)
```

Hmm... alangkah baiknya jika Anda menggunakan tipe tabel InnoDB untuk merancang tabel-tabel di atas supaya terdapat referential integration di antara setiap datanya.

Sedangkan untuk sampel recordnya, kita gunakan data seperti di bawah ini

```
INSERT INTO mhs VALUES ('M002', 'DWI AMALIA FITRIANI');
INSERT INTO mhs VALUES ('M003', 'FAZA FAUZAN');
```

```

INSERT INTO mk VALUES ('MK01', 'DATABASE', 3);
INSERT INTO mk VALUES ('MK02', 'PEMROGRAMAN DASAR', 2);
INSERT INTO mk VALUES ('MK03', 'PEMROGRAMAN WEB', 4);
INSERT INTO mk VALUES ('MK04', 'OOP', 3);
INSERT INTO mk VALUES ('MK05', 'PEMROGRAMAN VISUAL', 3);
INSERT INTO ambilmk VALUES ('M002', 'MK01', 3, 'GANJIL', '2008/2009');
INSERT INTO ambilmk VALUES ('M002', 'MK02', 2, 'GANJIL', '2008/2009');
INSERT INTO ambilmk VALUES ('M002', 'MK03', 2, 'GENAP', '2008/2009');
INSERT INTO ambilmk VALUES ('M002', 'MK04', 2, 'GENAP', '2008/2009');
INSERT INTO ambilmk VALUES ('M002', 'MK05', 4, 'GANJIL', '2009/2010');
INSERT INTO ambilmk VALUES ('M003', 'MK01', 2, 'GANJIL', '2008/2009');
INSERT INTO ambilmk VALUES ('M003', 'MK02', 3, 'GANJIL', '2008/2009');
INSERT INTO ambilmk VALUES ('M003', 'MK03', 3, 'GENAP', '2008/2009');
INSERT INTO ambilmk VALUES ('M003', 'MK04', 2, 'GENAP', '2008/2009');
INSERT INTO ambilmk VALUES ('M003', 'MK05', 3, 'GANJIL', '2009/2010');

```

OK.. tabel dan sampel record sudah ready, selanjutnya kita mulai membuat script untuk menampilkan list mahasiswanya terlebih dahulu.

### mhs.php

```

<?php

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

echo "<h1>Data Mahasiswa</h1>";

$query = "SELECT * FROM mhs";
$hasil = mysql_query($query);

echo "<table border='1'>";
echo "<tr><th>NIM</th><th>Nama Mhs</th><th>Statistik IP</th></tr>";
while ($data = mysql_fetch_array($hasil))
{
    echo "<tr>
        <td>".$data['nim']. "</td>
        <td>".$data['nama']. "</td>
        <td><a href='statistik.php?nim=".$data['nim']."'>View</a></td>
    </tr>";
}
echo "</table>";

?>

```

Perhatikan script mhs.php di atas! Untuk menampilkan list mahasiswa cukup membuat query SQL yang memanfaatkan tabel ‘mhs’. Kemudian di setiap baris data mahasiswa dalam tabel, kita buat link ‘VIEW’ yang diarahkan ke script ’statistik.php’. Link ke script ini kita sertakan parameter ’?nim=...’ karena NIM ini nanti digunakan sebagai acuan untuk membuat grafik IP si mahasiswa yang memiliki NIM tersebut. Hasil dari script mhs.php ini tampak seperti pada gambar pertama di atas.

Selanjutnya kita membuat script statistik.php nya

### statistik.php

```
<?php

mysql_connect ("dbhost", "dbuser", "dbpass");
mysql_select_db ("dbname");

// membaca nim dari parameter
$nim = $_GET['nim'];

// query untuk menampilkan NIM dan Nama Mahasiswa berdasarkan NIM
$query = "SELECT * FROM mhs WHERE nim = '$nim'";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

echo "<h1>Statistik Indeks Prestasi</h1>";

echo "<table>";
echo "<tr><td>NIM</td><td>:</td><td>".$data['nim']."</td></tr>";
echo "<tr><td>Nama Mhs</td><td>:</td><td>".$data['nama']."</td></tr>";
echo "</table>";

// menampilkan image grafik IP dari mahasiswa berdasarkan NIM
echo "<img src='stat.php?nim=".$nim."'>";

?>
```

Script statistik.php di atas pada prinsipnya adalah menggabungkan antara informasi si mahasiswa (NIM dan Nama) dengan grafik IP nya. Untuk menampilkan informasi si mahasiswa, cukup kita buat query SQL `SELECT * FROM mhs WHERE nim = '$nim'`, dan grafiknya kita embed menggunakan tag `<img src='stat.php?nim=".$nim."'>`. Script 'stat.php' adalah script untuk membuat grafiknya. Mengapa dalam tag `<img>` ini perlu ada parameter '?nim=...' nya? Ya... karena NIM itu nanti juga digunakan untuk acuan dalam membuat script grafik IP dari mahasiswa tersebut.

Mungkin Anda bertanya, mengapa tidak dijadikan satu saja antara script untuk menampilkan informasi mahasiswa dengan script untuk grafiknya? He.. 3x jawabannya adalah 'tidak bisa'. Karena script stat.php ini akan menghasilkan output berupa image. Sedangkan informasi mahasiswanya (statistik.php) outputnya berupa dokumen HTML sehingga keduanya tidak bisa langsung dipadukan menjadi satu. Solusinya adalah mengembed script grafiknya ke dalam script dokumen HTML menggunakan tag ``.

Trus... kita akan membuat script grafiknya. Untuk grafik, kita akan menggunakan JpGraph karena free and mudah digunakan 😊. Mau menggunakan Open Flash Chart nya yang digunakan KPU, gak jadi, coz it's very ribet 😱. Ane pengin yang simpel-simpel aja deh.

Untuk grafiknya, seperti yang kita lihat pada sampel di atas, akan kita pilih grafik yang merupakan gabungan dalam bentuk batang (bar chart) dan garis (line chart).

## stat.php

```
<?php

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

// membaca parameter NIM dari 
$nim = $_GET['nim'];

// menyiapkan array untuk menyimpan smt dan tahun
$smtThn = array();

// menyiapkan array untuk menyimpan ip
$ip = array();

// memanggil modul JpGraph untuk membuat grafik batang dan garis
// modul ini terletak dalam folder bernama 'modul'
include ("modul/jpgraph.php");
include ("modul/jpgraph_bar.php");
include ("modul/jpgraph_line.php");

// query sql untuk mendapatkan IP setiap semester dari mahasiswa
$query = "SELECT smt, tahun, sum(nilai * sks)/sum(sks) as ip
          FROM ambilmk, mk
          WHERE ambilmk.kodemk = mk.kodemk AND ambilmk.nim = '$nim'
          GROUP BY tahun, smt";

$hasil = mysql_query($query);
while ($data = mysql_fetch_array($hasil))
{
    // menyimpan data semester dan tahun hasil query ke dalam array $smtThn
    // smt dan tahun digabung dalam satu string untuk ditampilkan dalam sumbu-y
    // grafik
    array_unshift($smtThn, $data['smt'] . " ." . $data['tahun']);

    // menyimpan data IP hasil query ke dalam array $ip
    array_unshift($ip, $data['ip']);
}

// membuat image ukuran 500 x 300 pixel
$graph = new Graph(500,300,"auto");

// membuat skala grafik. Nilai 4 di sini adalah nilai maksimum sumbu Y nya,
// mengingat IP maks adalah 4
$graph->SetScale("textlin", 1, 4);

// membuat bayangan dari image
$graph->SetShadow();

// mengatur batas margin grafik
$graph->SetMargin(50,50,40,40);

// membuat bar plot dari data IP
$barplot = new BarPlot($ip);
```

```

// membuat line plot dari data IP
$lineplot=new LinePlot($ip);

// memberi warna merah pada bar plot
$barplot->SetFillColor("red");

// menampilkan value IP pada setiap bar
$barplot->value->show();

// mengatur tampilan value IP dengan format 1 digit desimal di belakang koma
$barplot ->value->SetFormat("%3.1f");

// mengatur ketebalan garis pada lineplot
$lineplot->SetWeight(3);

// mengatur posisi ujung line plot supaya terletak di tengah-tengah bar
$lineplot->SetBarCenter();

// menampilkan barplot ke dalam image
$graph->Add($barplot);

// menampilkan lineplot ke dalam image
$graph->Add($lineplot);

// menampilkan smt dan tahun pada sumbu X
$graph->xaxis-> SetTickLabels($smtThn);

// menampilkan title grafik
$graph->title->Set("Grafik Indeks Prestasi Mahasiswa");

// memberi label pada sumbu X
$graph->xaxis->title->Set("Semester");

// memberi label pada sumbu Y
$graph->yaxis->title->Set("Indeks Prestasi (IP)");

// mengatur jenis font pada title, label sumbu X dan label sumbu Y
$graph->title->SetFont(FF_FONT1, FS_BOLD);
$graph->yaxis->title->SetFont(FF_FONT1, FS_BOLD);
$graph->xaxis->title->SetFont(FF_FONT1, FS_BOLD);

// menampilkan output grafik
$graph->Stroke();
?>

```

#### **Catatan:**

Query SQL untuk menghasilkan data IP setiap semester dari mahasiswa untuk ditampilkan ke dalam grafik, sama dengan query SQL yang pernah dibahas pada [artikel sebelumnya](#)<sup>[1]</sup>.

Jika Anda ingin mendapatkan script di atas sekaligus modul JpGraph nya, maka dapat didownload di bawah ini.

#### **Download Script**

## 50. Beberapa Function PHP Untuk Keperluan File System

Kali ini saya akan mencoba menjelaskan beberapa function PHP yang dapat digunakan untuk keperluan manajemen file system. Beberapa di antaranya adalah function untuk mengkopi file, menghapus file, melihat total space atau free space yang ada dalam suatu direktori dan lain-lain.

Mudah-mudahan bermanfaat bagi Anda.

### **basename()**

Function ini digunakan untuk membaca nama file dari suatu path.

Contoh:

```
<?php
$path = "/home/httpd/html/index.php";
echo basename($path); // menghasilkan 'index.php'
echo basename($path, ".php") // menghasilkan 'index' (menghapus ekstensi
.php)
?>
```

Contoh lain, bila script di bawah ini disimpan dengan nama myscript.php

myscript.php

```
<?php
echo basename($_SERVER['PHP_SELF'], ".php"); // menghasilkan 'myscript'
?>
```

### **copy()**

Function ini digunakan untuk mengcopy file.

Contoh:

```
<?php
$pathAwal = "c:/windows/file.txt";
$pathTujuan = "d:/dataku/file.txt";
copy($pathAwal, $pathTujuan); // mengcopy file dari path awal ke path tujuan
?>
```

### **unlink()**

Function ini digunakan untuk menghapus file

Contoh:

```
<?php
```

```
$pathFile = "c:/dataku/data.dat";
unlink($pathFile); // menghapus file sesuai yang ada di path file
?>
```

### **rmdir()**

Function ini digunakan untuk menghapus direktori atau folder

```
<?php
$pathDirektori = "c:/dataku";
rmdir($pathDirektori); // menghapus folder 'dataku' yang ada di c:/
?>
```

### **disk\_free\_space()**

Function ini digunakan untuk membaca jumlah space yang free atau masih kosong dalam suatu direktori disk.

Contoh:

```
<?php
echo disk_free_space("c:"); // menampilkan jumlah free space yang ada di C:
(OS Windows)
echo disk_free_space("/"); // menampilkan jumlah free space yang ada di "/"
(OS Linux/Unix)
?>
```

### **disk\_total\_space()**

Function ini digunakan untuk membaca total space yang ada dalam suatu direktori disk.

Contoh:

```
<?php
echo disk_total_space("d:");
echo disk_total_space("/");
?>
```

### **file\_exists()**

Function ini digunakan untuk mengecek keberadaan suatu file atau direktori

```
<?php
// script ini digunakan untuk mengetahui ada tidaknya file bernama fileku.txt
dalam c:/dataku
$pathFile = "c:/dataku/fileku.txt";
if (file_exists($pathFile))
{
    echo "File $pathFile ada";
}
else {
```

```
        echo "File $pathFile tidak ada";
    }
?>
```

### **filesize()**

Function ini digunakan untuk membaca ukuran suatu file (dalam bytes)

Contoh:

```
<?php
$file = "c:/dataku/data.dat";
echo filesize($file);
?>
```

### **fileperms()**

Function ini menampilkan permission dari suatu file atau direktori.

Contoh:

```
<?php
$path = "/home/htdocs/www";
echo substr(sprintf('%o', fileperms($path)), -4); // menampilkan 0755 (nilai
chmod)
?>
```

### **rename()**

Function ini digunakan untuk mengubah nama file atau direktori

Contoh:

```
<?php
$fileAwal = "c:/data.txt";
$fileBaru = "c:/data2.txt";
rename($fileAwal, $fileBaru); // mengubah nama file data.txt menjadi
data2.txt yang berada di c:/
?>
```

### **rmdir()**

Function ini digunakan untuk menghapus direktori atau folder

Contoh:

```
<?php
$dirpath = "c:/data/sampah";
rmdir($dirpath); // menghapus folder 'sampah' yang berada di direktori
c:/data
```

```
?>
```

## **readir()**

Function ini digunakan untuk membaca isi sebuah direktori.

Contoh:

```
<?php
// script ini digunakan untuk menampilkan isi dari c:/
if ($handle = opendir("c:/")) {
    while ($false !== ($file = readdir($handle))) {
        echo $file."<br>";
    }
    closedir($handle);
}
?>
```

OK deh, demikian beberapa function yang bisa kita gunakan untuk file system processing. Mudah-mudahan bermanfaat bagi Anda penikmat PHP, syukur-syukur ada yang membuat script PHP yang berfungsi sebagai File Manager sendiri 😊

## 51. Teknik Scripting Mencegah SQL Injection di PHP

Serangan SQL injection seakan tidak ada matinya. Tiap hari selalu saja ada situs yang terkena serangan ini. Namun... semoga para hacker yang melakukan injection ini pada baik-baik semua. Mereka tidak hanya sekedar membobol, namun juga diharapkan memberitahukan kelemahan sistem situs pada si webmaster dan juga tidak mengacak-acak isi situs.

SQL injection merupakan teknik yang dilakukan para hacker untuk masuk ke sebuah sistem atau situs dengan memberikan perintah atau query SQL melalui metode GET atau POST memanfaatkan kelemahan sistem. Kelemahan sistem di sini bisa berasal dari script atau programnya sendiri atau dari dalam server.

Pada kesempatan ini saya akan coba bahas bagaimana teknik scripting untuk mencegah SQL injection melalui GET method. Artikel ini merupakan request dari mas Rosid Anyway.. thanks mas atas requestnya, karena hal ini membuat saya harus belajar dan cari akal sebelum membuat postingan ini 😊

OK... sebelum saya menulis postingan ini, saya sempatkan sebentar untuk mencari informasi seputar teknik-teknik yang biasa dilakukan untuk SQL injection ini. Setelah browsing beberapa saat... akhirnya saya menemukan beberapa artikel teknik SQL injection yang dijelaskan sangat detail. Selain itu saya juga iseng-iseng cari tutorial videonya. Dari yang saya temukan, ada banyak teknik SQL injection yang bisa dilakukan. Namun, dalam hal ini saya batasi pembahasan hanya pada teknik SQL injection melalui GET method atau melalui URL.

Biasanya, serangan SQL injection yang dilancarkan ke sebuah situs korban melalui URL berparameter, misalkan seperti ini:

```
http://situskorban/artikel.php?id=10
```

URL di atas memiliki parameter id dan biasanya diikuti dengan angka tertentu. Angka ini menunjukkan nomor id dari artikel yang sedang tampil.

Hanya dengan menambahkan beberapa query SQL di belakang parameter tersebut, seorang hacker mampu mengakses database yang digunakan oleh situs tersebut. Berikut ini salah satu contoh query yang diinjectkan melalui URL berparameter

```
http://situskorban/artikel.php?id=10 union all select  
1,concat(user,0x3a,pass,0x3a,email) from users --
```

Seorang hacker dalam melancarkan aksinya, biasanya butuh beberapa kali submit URL dan query sebelum akhirnya mendapatkan targetnya.

Nah... sekarang saya akan paparkan sedikit tentang bagaimana menghindari injection seperti halnya di atas. Point pentingnya di sini adalah bagaimana supaya parameter hanya bisa membaca nomor unik id dari artikelnya saja, dengan mengabaikan query yang disisipkan hacker.

Atau dengan kata lain bagaimana supaya parameter yang berisi query injection dihilangkan, misal apabila diberikan URL berikut ini

```
http://situskorban/artikel.php?id=10 union all select  
1,concat(user,0x3a,pass,0x3a,email) from users --
```

akan menjadi

```
http://situskorban/artikel.php?id=10
```

(bagian ‘union all select 1,concat(user,0×3a,pass,0×3a,email) from users –’ diabaikan)

OK... untuk menghilangkan query yang disisipkan pada parameter dalam URL, kita bisa menggunakan ‘teknik casting’. Dalam hal ini, kita mengcasting nilai parameter ke dalam tipe data integer. Sebagai contoh, misalkan kita memiliki sebuah string id = ‘10 union all select 1,concat(user,0×3a,pass,0×3a,email) from users –’. Apabila id ini dicasting ke dalam integer maka akan dihasilkan id = 10, dengan cara memberikan perintah id = (int) \$id.

```
<?php  
  
$id = "10 union all select 1,concat(user,0x3a,pass,0x3a,email) from users --  
";  
$id = (int) $id;  
echo $id;  
  
?>
```

Script di atas akan menghasilkan 10.

Nah... pada URL `http://situskorban/artikel.php?id=...`, biasanya dalam script artikel.php ini, pastilah di dalamnya terdapat perintah berikut

```
<?php  
  
$id = $_GET['id'];  
  
/* script untuk menampilkan artikel dengan id tertentu*/  
  
?>
```

Perintah `$id = $_GET['id'];` digunakan untuk membaca nilai parameter id-nya untuk kemudian ditampilkan artikelnya berdasarkan id tersebut. Nah.. teknik casting di atas dapat diterapkan pada script artikel.php ini

```
<?php  
  
$id = (int) $_GET['id'];  
  
/* script untuk menampilkan artikel dengan id tertentu*/  
  
?>
```

Lebih baik lagi, kalau kita tambahkan function absolut abs()

```
<?php  
$id = abs((int) $_GET['id']);  
/* script untuk menampilkan artikel dengan id tertentu*/  
?>
```

Mengapa perlu ditambahkan abs()? ya... karena ada pula teknik SQL injection yang memanfaatkan ‘negative number’ pada parameter URLnya, misalkan

```
http://situskorban/artikel.php?id=-10 union all select  
1,concat(user,0x3a,pass,0x3a,email) from users --
```

Di sini, function abs() hanya untuk menjamin saja bahwa nilai parameter id adalah bernilai positif.

Oya... Teknik di atas dapat juga diterapkan pada POST method melalui form.

OK... demikian tipsnya. Mudah-mudahan ada manfaatnya buat Anda. Oya... disamping teknik pencegahan dapat dilakukan dari sisi scripting seperti yang telah saya paparkan di atas, terdapat pula teknik pencegahan dari sisi server yang justru lebih disarankan. Salah satunya adalah dengan memberikan pengamanan [web server \(Apache\) dengan mod\\_security](#)<sup>[2]</sup>.

## 52. Mengenkripsi Parameter GET Method Untuk Keamanan

GET method adalah jenis request yang sering disalahgunakan oleh orang. Jenis request inilah yang kerap kali digunakan orang untuk menginjeksi suatu perintah atau kode yang tujuannya untuk merusak. Sebut saja misalkan teknik XSS (Cross Site Scripting), SQL injection, dan bahkan spam pun bisa masuk ke dalam buku tamu Anda atau sejenisnya melalui GET method ini.

Beberapa teknik untuk mencegah injection yang telah disebutkan di atas, seperti [sql injection](#)<sup>[1]</sup>, memang telah banyak digunakan. Namun, ada baiknya kita mengantisipasi hal-hal di atas dengan menciptakan suatu URL yang aman melalui teknik enkripsi pada parameter GET methodnya.

Simak lebih lanjut artikel ini untuk mengetahui caranya.

Apa sih request yang berjenis GET method itu? Ya.. GET method adalah request melalui URL atau link. Jenis request ini akan menyebutkan secara eksplisit parameter yang akan diproses pada script tujuan melalui URLnya. Perhatikan contoh URL berikut ini yang akan merequest menggunakan GET method untuk menghitung jumlah dari dua buah bilangan 10 dan 20.

`http://namasitus/jumlahkan.php?bil1=10&bil2=20`

Dari contoh URL di atas tampak bahwa script jumlahkan.php membutuhkan 2 parameter untuk memproses. Parameter pertama adalah ‘bil1’ yang diberi nilai 10 dan parameter kedua adalah ‘bil2’ yang diberi nilai 20. Nah.. dari sini tampak bahwa kedua parameter tertulis secara eksplisit dalam URL. Hal inilah yang nantinya akan dimanfaatkan orang untuk melakukan injection pada script jumlahkan.php ini.

Trus... bagaimana donk supaya aman, artinya supaya orang tidak tahu nama parameter dan valuenya dari URL yang berasal dari request GET method ini? Ya... jawabannya adalah menggunakan enkripsi. Dalam hal ini hanya parameter dan valuenya saja yang kita enkripsi.

OK.. deh.. gunakan enkripsi, trus... idenya dalam implementasi bagaimana? Ya.. step pertama adalah kita enkripsi terlebih dahulu parameter dan valuenya. Selanjutnya di script pemrosesannya, parameter dan value kita dekrip menjadi parameter dan value yang asli. Nah... barulah parameter dan value itu diproses sebagai input di dalam script pemrosesannya.

The next question... untuk mengenkripsi parameter ini menggunakan metode enkripsi apa? Ya.. Anda bisa menggunakan algoritma enkripsi apapun, asal Anda juga harus bisa melakukan proses dekripsinya atau membuat script untuk proses dekripsinya. Atau dengan kata lain, jangan gunakan algoritma enkripsi yang Anda tidak tahu cara mendekripsinya, contohnya md5() 😊 karena saat ini sepertinya belum ada orang yang bisa membuat dekripsinya.

Bagi Anda yang tidak atau belum mengenal tentang teknik enkripsi, Anda bisa mencoba script yang saya buat untuk proses enkripsi, dekripsi serta melakukan proses decoding. Proses decoding di sini diperlukan untuk membaca setiap nilai parameter dan valuenya.

Dalam script yang saya buat ini, saya gunakan algoritma AES (Advanced Encryption Standard). Adapun modul untuk proses enkripsi dan dekripsi ini saya manfaatkan class yang berasal dari [phpclasses.org](http://phpclasses.org)<sup>[2]</sup>.

Modul enkripsi dan dekripsi yang berasal dari [phpclasses.org](http://phpclasses.org)<sup>[2]</sup> sayangnya hanya bisa mengenkripsi dan mendekripsi 16 karakter string saja. Hal ini karena AES pada dasarnya mengenkripsi suatu string dengan panjang 16 karakter. Namun.. mengingat panjang parameter bisa jadi sangat panjang, maka script tersebut saya modifikasi sampai panjang karakter string tak terhingga 😊 Alhamdulillah saya bisa memodifikasinya.

### [ [Download Modul](#) ]

OK... trus.. bagaimana cara menggunakan modul tersebut? Nah.. saya ambil contoh aplikasi sederhana yaitu menjumlahkan dan membandingkan dua bilangan menggunakan request GET method.

Misalkan Anda punya script yang di dalamnya terdapat link untuk melakukan request GET method seperti di bawah ini

```
<a href="proses.php?bil1=17&bil2=50&action=jumlahkan">Jumlahkan</a>
```

dan

```
<a href="proses.php?bil1=166&bil2=150&action=bandingkan">Bandingkan</a>
```

Nah... untuk mengenkripsi parameternya, gunakan function paramEncrypt(). Contoh:

```
<?php
include "function.php";
echo "<p><a
href='proses.php?' . paramEncrypt ('bil1=17&bil2=50&action=jumlahkan') . "'>Jumlah
kan</a></p>";
echo "<p><a
href='proses.php?' . paramEncrypt ('bil1=166&bil2=150&action=bandingkan') . "'>Ban
dingkan</a></p>";
?>
```

Keterangan:

Script function.php ini harus diincludekan setiap kali perintah paramEncrypt() ini akan digunakan.

Bila script di atas Anda jalankan di browser maka link berparameter tersebut akan berbentuk

```
<p><a
href='proses.php?b19a49e6951a7f9eea3edf67f04422e3c018b3faa88b027d1c7df08c6bc0
2e33'>Jumlahkan</a></p>
```

```
<p><a href='proses.php?8bb81c11c862bf6a435ad5c5dcaf79729469d5f55d02fce77af7d52ed08cc4a2fe6a4a4fd19a52fc7cbf06d24d0e9e'>Bandingkan</a></p>
```

Hmm... tidak bisa dibaca bukan nama-nama parameter dan valuenya? 😊

Nah... selanjutnya kita buat script proses.php yang di dalamnya terdapat proses dekripsi dan decoding.

```
<?php

include "function.php";

// mendecode parameter dan value sekaligus proses dekripsinya
// hasil decoding parameter disimpan dalam bentuk array, misalnya $var

$var = decode($_SERVER['REQUEST_URI']);

// untuk membaca value dari parameter tertentu gunakan $var['nama parameter']

if ($var['action'] == "jumlahkan")
{
    // proses menjumlahkan

    $hasil = $var['bill'] + $var['bil2'];
    echo "Hasil penjumlahan ".$var['bill']."' dan ".$var['bil2']."' adalah : ".
        $hasil;
}
else if ($var['action'] == "bandingkan")
{
    // proses perbandingan

    if ($var['bill'] > $var['bil2']) echo $var['bill']."' lebih besar
dari ".$var['bil2'];
    else if ($var['bill'] < $var['bil2']) echo $var['bill']."' lebih
kecil dari ".$var['bil2'];
    else echo $var['bill']."' sama dengan ".$var['bil2'];
}
?>
```

Nah... mudah bukan menggunakannya. Maaf... dalam hal ini saya tidak menguraikan teknik enkripsi, dekripsi dan decodingnya karena algoritmanya sangat panjang.

Selain untuk faktor keamanan, biasanya enkripsi parameter ini juga digunakan dalam program afiliasi marketing, PPC (pay per click) seperti Google Adsense, dan advertising program dalam Facebook. Dalam internet marketing, enkripsi parameter digunakan untuk mengaburkan id affiliate atau url iklan supaya sulit diingat dan mencegah by pass menuju situs tujuan iklan tanpa mengklik iklan tersebut. Wah pintar juga ya para pembuat affiliate program ini? 😊

### 53. Membuat Sistem Login Aplikasi Multi User

Dalam suatu aplikasi bisa jadi terdapat lebih dari satu user dan dengan level yang berbeda-beda. Dengan adanya pembagian level tersebut diharapkan adanya hak akses yang berbeda. Untuk memudahkan Anda dalam membayangkan hal ini, misalkan terdapat 5 user katakanlah A, B, C, D dan E dimana user A, B dan C adalah level ‘administrator’, sedangkan D dan E user biasa. Nah... user A, B, dan C ini misalkan berhak mengakses halaman 1, 2 dan 3 dari aplikasi sedangkan D dan E hanya boleh mengakses halaman 3 saja. Dengan kata lain hak akses ke halaman-halaman tersebut ditentukan oleh level usernya.

Pemberian hak akses pada masing-masing user ini bisa dalam bentuk link menu. Ketika seorang user telah melakukan login, dan loginnya sukses maka akan muncul menu link ke halaman-halaman yang boleh diaksesnya. Sebagai gambaran misalnya pada contoh di atas, setelah user A login dan loginnya berhasil maka akan muncul menu link menuju ke halaman 1, 2 dan 3 karena si A ini levelnya Administrator. Namun untuk si D, karena levelnya user biasa, maka hanya akan muncul menu link menuju halaman 3 saja.

Nah... lantas bagaimana ide dan teknik dalam membuat sistem login terkait dengan hak akses untuk multi user seperti kasus di atas? Inilah yang akan saya berikan rahasianya dalam artikel ini. Artikel ini merupakan request dari xfreecode sekaligus salah satu dari sekian banyak trik dan teknik yang saya paparkan pada peserta [kursus online PHP](#)<sup>[2]</sup> bersama saya.

Ide dan teknik untuk menyelesaikan kasus di atas sebenarnya hanyalah pada penggunaan SESSION. Setelah user berhasil login, maka dibaca level usernya. Untuk memudahkan pengecekan level user ini, maka level user disimpan pada SESSION. Nah.. level user yang tersimpan dalam SESSION inilah yang nantinya akan banyak dicek untuk keperluan pembatasan hak akses user pada halaman-halaman yang sifatnya private.

OK.. OK... sudah paham deh idenya, tapi gini... bagaimana seandainya ada user yang nakal? misalkan user E dengan level user biasa kebetulan tahu link URL menuju ke halaman 1. Lantas si E ini mencoba mem bypass link menuju halaman 1 melalui URL langsung tanpa melewati menu link, padahal halaman 1 ini adalah restricted area bagi user biasa (dalam hal ini si E). Nah lo... gimana hayoo?? ya... seorang programmer harus pinter-pinter ngakali. Trus... Gimana ngakalinnya? ya... sebelum bagian script yang menampilkan konten pada halaman 1 terlebih dahulu harus mengecek level usernya sebagai ‘administrator’ atau tidak, melalui pengecekan SESSION nya. Jika ya, maka tampilkan kontennya, dan jika tidak maka munculkan pesan ‘Anda mau mencoba masuk ya?? Kagak boleh atuh’ 😛

OK.. deh ... stop berteori, sekarang let's do it! 😊

Untuk studi kasusnya, misalkan kita punya data user sebagaimana tersaji dalam tabel berikut ini

```
CREATE TABLE user (
    username varchar(20),
    password varchar(20),
    level varchar(20),
    PRIMARY KEY(username)
```

```

);
INSERT INTO `user` VALUES ('a', 'pass1', 'admin');
INSERT INTO `user` VALUES ('b', 'pass2', 'admin');
INSERT INTO `user` VALUES ('c', 'pass3', 'user');
INSERT INTO `user` VALUES ('d', 'pass4', 'user');
INSERT INTO `user` VALUES ('e', 'pass5', 'user');

```

Wah.. passwordnya kok tidak dienkripsi? he 3x... ya itu saya sengaja karena supaya Anda bisa mencoba script yang saya buat ini. Memang sebaiknya passwordnya Anda enkripsi, paling tidak menggunakan [md5\(\)](#)<sup>[3]</sup>.

Nah.. selanjutnya kita buat form loginnya.

### login.htm

```

<form method="post" action="loginsubmit.php">
  <table border="0">
    <tr>
      <td>Masukkan Username </td>
      <td><input name="username" type="text"></td>
    </tr>
    <tr>
      <td>Masukkan Password </td>
      <td><input name="pass" type="password"></td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td><input type="submit" name="Submit" value="Submit"></td>
    </tr>
  </table>
</form>

```

Form login sudah dibuat, berikutnya kita buat script untuk memproses loginnya.

### loginsubmit.php

```

<?php

// memulai session
session_start();

mysql_connect('dbhost', 'dbuser', 'dbpass');
mysql_select_db('dbname');

$username = $_POST['username'];
$password = $_POST['pass'];

// query untuk mendapatkan record dari username
$query = "SELECT * FROM user WHERE username = '$username'";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

// cek kesesuaian password
if ($password == $data['password'])

```

```

{
    echo "<h1>Login Sukses</h1>";

    // menyimpan username dan level ke dalam session
    $_SESSION['level'] = $data['level'];
    $_SESSION['username'] = $data['username'];

    // tampilkan menu
    include "menu.php";
}

else echo "<h1>Login gagal</h1>";

?>

```

Bila Anda perhatikan pada script di atas, maka terlihat bahwa menunya membaca script menu.php. Di dalam script menu.php sendiri nantinya terdapat pengecekan level usernya yang tadi tersimpan dalam session sewaktu login sukses. Bila level usernya adalah ‘admin’ maka tampilkan menu link menuju halaman 1, 2, dan 3. Begitu pula untuk level ‘user’.

### **menu.php**

```

<?php

session_start();

if ($_SESSION['level'] == "admin")
{
    // tampilkan menu untuk admin
    echo "<a href='hal1.php'>Halaman 1</a> ";
    echo "<a href='hal2.php'>Halaman 2</a> ";
    echo "<a href='hal3.php'>Halaman 3</a> ";
    echo "<a href='logout.php'>Logout</a> ";
}
else if ($_SESSION['level'] == "user")
{
    // tampilkan menu untuk user biasa
    echo "<a href='hal3.php'>Halaman 3</a>";
    echo "<a href='logout.php'>Logout</a> ";
}

?>

```

Selanjutnya, kita bisa mulai membuat konten pada halaman 1, 2 dan 3. Ingat.. sebelum menampilkan kontennya sebaiknya dicek level usernya terlebih dahulu. Bila levelnya tidak sesuai dengan hak aksesnya, maka jangan ragu untuk menolak 😊 Untuk halaman 1, dari kasus yang kita ambil hanya bisa diakses oleh user berlevel ‘admin’ saja. Begitu pula untuk halaman 2. Namun untuk halaman 3, baik level ‘admin’ maupun ‘user’ biasa boleh mengaksesnya.

### **hal1.php**

```
<?php
```

```

session_start();

// cek apakah user yang mengakses halaman ini sudah melalui login atau belum
// logikanya jika user telah login dan sukses, maka SESSION level dan SESSION
username ini pasti sudah ada
// jika ada user yang mencoba akses halaman ini tanpa login, maka logikanya
kedua SESSION belum ada

if (isset($_SESSION['level']) && isset($_SESSION['username']))
{
    // tampilkan menu.
    // menu hanya ditampilkan bila halaman ini diakses oleh user yang telah
login

    include "menu.php";

    // cek level user apakah admin atau bukan

    if ($_SESSION['level'] == "admin")
    {
        // tampilkan konten halaman 1 jika levelnya admin

        echo "<h1>Halaman 1</h1>";
        echo "<p>Selamat datang di halaman satu wahai user :";
        echo ". $_SESSION['username'] . "</p>";
        echo "<p>Ini adalah konten halaman 1</p>";
    }
    else
    {
        // jika levelnya bukan admin, tampilkan pesan
        echo "<h1>Maaf.. Anda bukan user berlevel Administrator</h1>";
    }
}
else
{
    echo "<h1>Nah.. Anda pasti mencoba akses halaman ini tanpa login
ya?</h1>";
}

?>

```

Untuk halaman 2, konsepnya juga sama seperti script halaman 1 di atas, hanya yang berbeda pada kontennya tentu saja 😊 Sekarang untuk halaman ke-3 di mana yang bisa mengakses adalah semua level user, scriptnya adalah:

### hal3.php

```

<?php

session_start();

if (isset($_SESSION['level']) && isset($_SESSION['username']))
{
    include "menu.php";

```

```

if (( $_SESSION['level'] == "admin" ) || ( $_SESSION['level'] == "user" ))
{
    echo "<h1>Halaman 3</h1>";
    echo "<p>Selamat datang di halaman tiga wahai user : ". $_SESSION['username']. "</p>";
    echo "<p>Ini adalah konten halaman 3</p>";
}
else
{
    echo "<h1>Nah.. Anda pasti mencoba akses halaman ini tanpa login ya?</h1>";
}

?>

```

Eits... hampir lupa, yang terakhir kita buat script logoutnya. Pada prinsipnya proses logout adalah menghapus semua session yang tadi terbentuk ketika login sukses.

### **logout.php**

```

<?php

session_start();
session_destroy();
echo "<h1>Anda sudah logout</h1>";
echo "<p><a href='formlogin.htm'>Login Kembali</a></p>";

?>

```

That's it... mudah bukan membuatnya? Mudah-mudahan artikel di atas menjawab semua unekeuneke yang dihadapi bagi Anda yang ingin membuat aplikasi dengan multi user.

[ [\*\*DOWNLOAD SCRIPT \[4\]\*\*](#) ]

## 54. Script PHP untuk Menampilkan Siswa Belum Bayar SPP (A Study Case)

Bila Anda merupakan pengunjung setia blog ini, maka Anda akan pernah mendapatkan [artikel yang hampir sama](#)<sup>[1]</sup> beberapa waktu yang lalu. Namun perbedaan artikel berikut ini adalah bagaimana cara menampilkan data siswa yang belum membayar SPP pada bulan dan tahun tertentu. Selain mengetahui name by name siswanya, juga akan ditampilkan tanggal SPP terakhir yang dibayarkan siswa yang menunggak tadi. Untuk mencari solusi dari hal ini, kita akan coba gunakan script PHP. Oya.. artikel ini merupakan request dari mas Hendriko, salah satu [member saya](#)<sup>[2]</sup>.

OK mas Hendriko dan pengunjung setia blog ini, untuk membuat script PHP seperti yang kita inginkan di atas, langkah pertama adalah kita cari idenya dahulu. Ini adalah keharusan sebelum memulai scripting. Bila kita tidak punya ide, maka... kita akan bingung sendiri. Gimana mau bisa buat script, kalo bingung gimana mau memulai scriptnya. So... guys... ‘ide’ adalah hal yang utama.

Lantas.. idenya bagaimana nih? OK... ide ini bisa diperoleh berdasarkan output yang kita inginkan. Output yang kita harapkan adalah menampilkan Nomor Induk Siswa (NIS) yang belum bayar SPP pada bulan dan tahun tertentu, namanya siapa, tanggal terakhir dia bayar SPP berapa, dan dia bayarkan SPP terakhir tersebut untuk bulan dan tahun berapa. OK.. itu outputnya.

Sekarang... idenya bagaimana untuk mendapatkan output di atas? OK... untuk mendapatkan NIS dan nama siswa yang belum membayar pada bulan dan tahun tertentu, kita bisa menggunakan perintah query SQL seperti halnya yang telah dijelaskan pada [artikel sebelumnya](#)<sup>[1]</sup>. Trus.. untuk mendapatkan tanggal terakhir SPP dibayarkan dan informasi lainnya, gimana caranya? Ya... thats the point. Mmm.. enaknya kita buat script PHP nya dulu untuk menampilkan NIS dan Nama Siswa yang belum bayar dulu yah.. informasi yang lainnya nyusul. OK.. ini dia script PHP untuk menampilkan NIS dan Nama Siswanya, dengan asumsi kita gunakan tabel yang sama seperti pada [artikel sebelumnya](#)<sup>[1]</sup>. Oya.. dalam contoh script ini akan ditampilkan NIS dan Nama Siswa yang belum bayar SPP pada bulan Februari 2008.

```
<?php

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

$query = "SELECT nis, namasiswa FROM siswa WHERE nis NOT IN (SELECT nis FROM
bayarspp WHERE byrbln = 2 AND byrthn = '2008')";
$hasil = mysql_query($query);
while ($data = mysql_fetch_array($hasil))
{
    $nis = $data['nis'];
    $namasiswa = $data['namasiswa'];

    echo $nis." ".$namasiswa."<br />";
}
?>
```

Perhatikan script di atas! Dalam script di atas kita gunakan query SQL yang sama seperti yang digunakan pada [artikel sebelumnya](#)<sup>[1]</sup>.

Next... bagaimana dengan informasi yang lainnya, seperti tanggal terakhir si siswa bayar SPP dan untuk bulan dan tahun berapa ia bayarkan. Nah.. untuk mencari informasi tersebut, kita gunakan query lagi memanfaatkan NIS yang kita telah peroleh dari query sebelumnya.

Query SQL untuk mencari tanggal terakhir seorang mahasiswa berNIS ‘X’ membayar SPP dan untuk bulan dan tanggal berapa ia bayarkan adalah dengan perintah sbb:

```
SELECT tglbyr, byrbln, byrthn FROM bayarspp WHERE nis = 'X' ORDER BY tglbyr DESC LIMIT 0, 1;
```

Nah.. NIS dalam klausula WHERE tersebut nantinya kita gantikan dengan NIS para siswa yang belum bayar SPP pada bulan Pebruari 2008 tersebut.

Oya... ide query SQL di atas adalah, kita sort semua data pembayaran yang telah dilakukan siswa berNIS ‘X’ berdasarkan tanggal bayar secara DESC. Untuk mendapatkan data pembayaran yang terakhir saja, maka kita gunakan LIMIT 0, 1. Selesai... 😊

Next... kita gabungkan query di atas dengan script PHP sebelumnya. Bagaimana menggabungkannya? dan dimana letaknya dalam script? Ya.. letaknya di dalam WHILE(). Karena dalam WHILE lah kita menampilkan data NIS dan Nama Siswa yang belum bayar SPP secara satu persatu.

```
<?php

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

$query = "SELECT nis, namasiswa FROM siswa WHERE nis NOT IN (SELECT nis FROM bayarspp WHERE byrbln = 2 AND byrthn = '2008')";
$hasil = mysql_query($query);
while ($data = mysql_fetch_array($hasil))
{
    $nis = $data['nis'];
    $namasiswa = $data['namasiswa'];

    $query2 = "SELECT tglbyr, byrbln, byrthn FROM bayarspp WHERE nis = '$nis' ORDER BY tglbyr DESC LIMIT 0, 1";
    $hasil2 = mysql_query($query2);
    $data2 = mysql_fetch_array($hasil2);

    echo $nis." ".$namasiswa." ".$data2['tglbyr']." ".$data2['byrbln']."'".
    ".$data2['byrthn']."'<br />";
}

?>
```

OK.. selesai deh.. eh tapi kok amburadul ya tampilannya? ya... gunakan tabel saja untuk mengatur tampilannya supaya cuantiq... 😊

```

<?php

mysql_connect("dbhost", "dbuser", "dbpass");
mysql_select_db("dbname");

$query = "SELECT nis, namasiswa FROM siswa WHERE nis NOT IN (SELECT nis FROM
bayarspp WHERE byrbln = 2 AND byrthn = '2008'))";
$hasil = mysql_query($query);

echo "<h3>Daftar siswa yang belum bayar SPP bulan Februari Th 2008</h3>";

echo "<table border='1'>";
echo "<tr><td>NIS</td><td>Nama Siswa</td><td>Tanggal Bayar SPP
Terakhir</td><td>Untuk Pembayaran Bulan</td><td>SPP Tahun</td></tr>";

while ($data = mysql_fetch_array($hasil))
{
    $nis = $data['nis'];
    $namasiswa = $data['namasiswa'];

    $query2 = "SELECT tglbyr, byrbln, byrthn FROM bayarspp WHERE nis = '$nis'
ORDER BY tglbyr DESC LIMIT 0, 1";
    $hasil2 = mysql_query($query2);
    $data2 = mysql_fetch_array($hasil2);

    echo
"<tr><td>".$nis."</td><td>".$namasiswa."</td><td>".$data2['tglbyr'].".</td><td>".$data2['byrbln'].".</td><td>".$data2['byrthn'].".</td></tr>";

}
echo "</table>";

?>

```

Nah.. Anda bisa lihat sendiri tampilannya sekarang bila script di atas ditampilkan ke browser.. betapa cantiqnya dia he 3x 😊 Supaya outputnya bisa banyak dan rame, Anda bisa memperbanyak variasi datanya terlebih dahulu terutama di tabel ‘bayarspp’. Atau kalau tidak ingin repot, Anda bisa gunakan data berikut ini:

```

INSERT INTO `bayarspp` VALUES ('10001', '2008-01-10', 1, '2008');
INSERT INTO `bayarspp` VALUES ('10002', '2008-01-12', 1, '2008');
INSERT INTO `bayarspp` VALUES ('10003', '2008-01-10', 1, '2008');
INSERT INTO `bayarspp` VALUES ('10001', '2008-02-09', 2, '2008');
INSERT INTO `bayarspp` VALUES ('10003', '2008-02-13', 2, '2008');
INSERT INTO `bayarspp` VALUES ('10004', '2008-01-11', 1, '2008');
INSERT INTO `bayarspp` VALUES ('10004', '2007-12-20', 12, '2007');
INSERT INTO `bayarspp` VALUES ('10005', '2007-12-20', 12, '2007');
INSERT INTO `bayarspp` VALUES ('10005', '2008-01-10', 1, '2008');
INSERT INTO `bayarspp` VALUES ('10005', '2007-11-20', 11, '2007');

```

Nah selamat mencoba dan berkreasi sendiri dengan scripting ya..

Oya... pesan saya terakhir, sebelum Anda memulai scripting, coba cari idenya dulu karena itu adalah ‘keharusan’.

## 55. Membuat Script Random Image dengan PHP

Kali ini saya akan coba memaparkan bagaimana ide dan cara membuat random image. Image yang kemunculannya akan dirandom ini nantinya disimpan ke dalam database, dan bukan berdiri sendiri sebagai file dalam direktori atau folder. Dalam hal ini user bisa mengupload image file ke server melalui form. Dengan di tempatkannya image ke dalam database, maka tingkat sekuriti dapat lebih aman karena kita tidak perlu meng-chmod 777 folder di server sebagai tempat image diupload. Selain itu... biasanya banyak hosting yang menyediakan fitur space MySQL yang unlimited, dibandingkan space quota di server yang biasanya terbatas 😊

Secara umum, ide untuk membuat random image ini sangat mudah. Konsepnya adalah sebagai berikut: user mengupload image dari komputer client ke server, tepatnya ke database. Selanjutnya untuk proses mengacak imagenya kita gunakan function `RAND()` yang ada di MySQL. Konsep ini hampir sama dengan [membuat random quotes](#)<sup>[1]</sup>. Tambahan lagi, untuk memunculkan image dari database ke browser, kita memerlukan script khusus. Script khusus apa nih maksudnya? Simak baik-baik artikel ini selanjutnya.

Seperti konsep yang telah kita tentukan di atas, pertama kali kita coba membuat fitur untuk upload image ke database terlebih dahulu. Nah... sekarang kita create terlebih dahulu tabelnya.

```
CREATE TABLE tabelimage
(
    id INT AUTO_INCREMENT,
    image LONGBLOB,
    PRIMARY KEY (id)
);
```

Perhatikan query di atas, bahwa untuk menyimpan data image kita perlu tipe data BLOB. Kalau tidak salah ada 3 jenis tipe data BLOB ini, yaitu TINYBLOB, MEDIUMBLOB dan LONGBLOB. Ketiganya dibedakan berdasarkan size data yang bisa tertampung. Selanjutnya, untuk ID, field ini diperlukan untuk pembeda antar image yang diupload. ID ini nanti yang akan dirandom oleh function `RAND()`.

OK.. next...

Sekarang kita coba buat script untuk upload imagenya.

### upload.htm

```
<h1>Upload Image</h1>
<form method="post" enctype="multipart/form-data" action="simpan.php">
Pilih image :
<input type="hidden" name="MAX_FILE_SIZE" value="100000"><br />
<input name="image" type="file">
<input type="submit" value="Submit" name="submit">
</form>
```

Perhatikan pada komponen <input type="hidden" name="MAX\_FILE\_SIZE" value="100000">. Atribut value="100000" ini menunjukkan maksimum size dari image yang bisa diupload adalah 100Kb. Anda bisa memperbesar valuenya. Tapi dengan catatan: value ini harus lebih kecil atau sama dengan nilai dalam parameter **upload\_max\_filesize** yang ada dalam konfigurasi PHP pada file **php.ini**. Default dari nilai parameter tersebut adalah 2Mb.

Selanjutnya kita buat script simpan.php nya.

### simpan.php

```
<?php

// membaca file yang diupload dan disimpan dalam TEMP
$tmpName = $_FILES['image']['tmp_name'];

// membuka isi file image yang disimpan dalam TEMP
$fp = fopen($tmpName, 'r');

// membaca isi file image yang disimpan dalam TEMP
// isi file image yang dibaca berupa string
$dataimage = fread($fp, filesize($tmpName));

// apabila ditemukan karakter slash pada string, maka akan diubah menjadi
// double slash
// prosedur ini bila tidak dilakukan akan merusak image
$dataimage = addslashes($dataimage);

// menutup file dalam TEMP yang telah dibaca
fclose($fp);

// koneksi ke MySQL
mysql_connect('hostName', 'dbUserName', 'dbUserPass');
mysql_select_db('dbName');

// query menyisipkan data image ke tabel
$query = "INSERT INTO tabelimage(image) VALUES ('$dataimage')";
$hasil = mysql_query($query);

// menampilkan konfirmasi sukses tidaknya upload
if ($hasil) echo "Upload image sukses";
else echo "Upload image gagal";

?>
```

Sekarang... kita buat script untuk menampilkan random imagennya. Dalam hal ini yang dirandom adalah ID dari imagennya.

### random.php

```
<?php

// koneksi ke mysql
mysql_connect('hostName', 'dbUserName', 'dbUserPass');
```

```

mysql_select_db('dbName');

// query untuk merandom ID dari image
$query = "SELECT id FROM tabelimage ORDER BY RAND()";
$hasil = mysql_query($query);

echo "<img src='showimage.php?id=".$data['id']."' />";

?>

```

Untuk bisa menampilkan image yang dirandom, kita perlu membuat script khusus (pada contoh ini adalah showimage.php). Nah... yang menjadi masalah sekarang adalah bagaimana mengaitkan antara ID image yang dirandom dengan image yang akan ditampilkan. Masalah ini bisa dipecahkan dengan menambahkan parameter `id` di belakang script showimage.php nya pada tag `<img>`.

OK... selanjutnya bagaimana isi script showimage.php nya. Ini dia...

### **showimage.php**

```

<?php

// koneksi ke mysql
mysql_connect('hostName', 'dbUserName', 'dbUserPass');
mysql_select_db('dbName');

// membaca nilai dari parameter ID yang berasal dari URL:
showimage.php?id=...

$id = $_GET['id'];

// menampilkan image berdasarkan ID imagennya

$query = "SELECT image FROM tabelimage WHERE id = $id";
$hasil = mysql_query($query);
$data = mysql_fetch_array($hasil);

// header untuk menampilkan image
header('Content-type: image/jpeg');

// menampilkan data imagennya
echo $data['image'];

?>

```

Yah... begitulah caranya membuat random image dengan PHP. Anda bisa menerapkan random image ini untuk membuat banner dalam rangka bisnis online Anda 😊 atau yang lain.

Semoga artikel ini ada manfaatnya buat semua.

## 56. Apa Sih Gunanya Session di PHP?

Apa sih gunanya session di PHP? Beberapa pengunjung setia blog saya ini menanyakan hal tersebut ke saya baik lewat email, YM atau sekedar komentar di blog ini. Nah... melalui tulisan inilah, saya akan berusaha untuk menjawabnya.

Secara umum, session digunakan untuk menyimpan suatu informasi antar proses request, baik request dalam bentuk POST atau GET. Bingung yah ?? He.. he.. he... OK saya akan ambil contoh untuk menggambarkan hal ini.

Salah satu contoh yang menggambarkan penggunaan session adalah proses login. Dalam hal ini user akan memasukkan usernamenya melalui form login. Setelah login berhasil, user tersebut dihadapkan pada link menu navigasi yang menuju ke beberapa halaman web. Nah... apabila kita ingin username tersebut akan selalu tampil atau tercatat di halaman-halaman web tersebut, maka username tadi haruslah disimpan dalam session.

Untuk memudahkan lagi pemahaman, silakan Anda buat script yang menggambarkan keadaan di atas.

Pertama-tama kita buat form login terlebih dahulu

### login.htm

```
<form method="post" action="submit.php">
  Username <input type="text" name="username">
  Password <input type="password" name="password">
  <input type="submit" name="submit" value="Submit">
</form>
```

Nah... selanjutnya kita buat script untuk mengolah proses login. Oya, dalam hal ini andaikan password login diabaikan dahulu ya... karena saya akan fokuskan pembahasan ke konsep session, bukan proses loginnya. Dengan arti lain, untuk contoh ini anggap saja proses loginnya sukses. Login dikatakan sukses bila password yang dimasukkan user yang bersangkutan ketika dalam form login sama dengan passwordnya yang tersimpan dalam aplikasi.

### submit.php

```
<?php
$namauser = $_POST['username'];
$password = $_POST['password'];

if (login sukses)
{
  echo "<p>Selamat datang ".$namauser."</p>";
  echo "<p>Berikut ini menu navigasi Anda</p>";
  echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";
}
```

```
?>
```

Dari script di atas tampak bahwa username akan muncul atau dikenal di halaman submit.php. Username ini akan ditampilkan di ucapan selamat datang. Mengapa username ini dikenal di halaman ini? Ya... karena halaman submit.php ini merupakan halaman tujuan langsung setelah proses request melalui form login. Selanjutnya perhatikan, bahwa setelah login sukses terdapat 3 link menuju ke suatu halaman tertentu. Harapan kita, di setiap halaman tersebut username akan selalu tercatat dan ditampilkan. OK... kita buat script untuk masing-masing halaman tersebut.

### hal1.php

```
<?php  
  
echo "<h1>Ini halaman pertama</h1>";  
echo "<p>Anda login sebagai ".$namauser."</p>";  
echo "<p>Berikut ini menu navigasi Anda</p>";  
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";  
  
?>
```

### hal2.php

```
<?php  
  
echo "<h1>Ini halaman kedua</h1>";  
echo "<p>Anda login sebagai ".$namauser."</p>";  
echo "<p>Berikut ini menu navigasi Anda</p>";  
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";  
  
?>
```

### hal3.php

```
<?php  
  
echo "<h1>Ini halaman ketiga</h1>";  
echo "<p>Anda login sebagai ".$namauser."</p>";  
echo "<p>Berikut ini menu navigasi Anda</p>";  
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";  
  
?>
```

Nah... Anda perhatikan bahwa dalam ketiga script masing-masing halaman tujuan, username akan ditampilkan dalam statu login. Namun apa yang terjadi bila Anda menuju ke halaman-halaman tersebut? Munculkah username tersebut? Bim... salabim... ternyata username tidak muncul. Sehingga dari contoh ini dapat kita simpulkan bahwa username ini hanya akan dikenal pada proses request pertama (login), setelah itu bila menuju ke halaman-halaman lain

pada link (proses request kedua, ketiga dst...) maka username tersebut tidak dikenali lagi. So... gimana donk, supaya username tersebut masih bisa dikenali? Yup... kita bisa menggunakan session untuk menyimpan username tersebut. Kenapa kok istilahnya ‘bisa’, bukannya ‘harus’ ? Ya... karena penggunaan session ini merupakan salah satu cara saja, cara yang lain Anda bisa menggunakan cookies.

OK... jadi kita bisa menggunakan session, lantas caranya bagaimana menyimpan username ini ke dalam session? Caranya adalah memberikan perintah berikut ini:

```
$_SESSION['namauser'] = $username;
```

Perintah di atas disisipkan pada script submit.php. Oya.. jangan lupa sebelum perintah tersebut diberikan, session harus dijalankan terlebih dahulu dengan perintah session\_start(). Sehingga isi dari script submit.php menjadi seperti ini

### **submit.php**

```
<?php
session_start();

$namauser = $_POST['username'];
$password = $_POST['password'];

if (login sukses)
{
    $_SESSION['namauser'] = $namauser;

    echo "<p>Selamat datang ".$namauser."</p>";
    echo "<p>Berikut ini menu navigasi Anda</p>";
    echo "<p><a href='hall1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";
}

?>
```

Secara umum, perintah untuk menyimpan nilai ke dalam session adalah sebagai berikut

```
$_SESSION['nama session'] = value;
```

Mmm... untuk nama session tidak boleh ada spasi. Kita tidak hanya bisa menyimpan suatu nilai berbentuk tunggal ke dalam session, namun bisa juga nilai berupa array.

Sekarang... bagaimana cara menampilkan nilai yang telah tersimpan dalam session? ya... caranya hanya dengan memanggil sessionnya. Berikut ini contoh untuk menampilkan username yang telah disimpan ke dalam session pada kasus di atas.

### **submit.php**

```
<?php
session_start();
```

```

$namauser = $_POST['username'];
$password = $_POST['password'];

if (login sukses)
{
$_SESSION['namauser'] = $namauser;

echo "<p>Selamat datang ".$_SESSION['namauser']."</p>";
echo "<p>Berikut ini menu navigasi Anda</p>";
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";
}

?>

```

Jangan lupa untuk menerapkan hal yang sama pada ketiga halaman lain.

### **hal1.php**

```

<?php
session_start();

echo "<h1>Ini halaman pertama</h1>";
echo "<p>Anda login sebagai ".$_SESSION['namauser']."</p>";
echo "<p>Berikut ini menu navigasi Anda</p>";
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";

?>

```

### **hal2.php**

```

<?php
session_start();

echo "<h1>Ini halaman kedua</h1>";
echo "<p>Anda login sebagai ".$_SESSION['namauser']."</p>";
echo "<p>Berikut ini menu navigasi Anda</p>";
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";

?>

```

### **hal3.php**

```

<?php
session_start();

echo "<h1>Ini halaman ketiga</h1>";
echo "<p>Anda login sebagai ".$_SESSION['namauser']."</p>";
echo "<p>Berikut ini menu navigasi Anda</p>";
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";

```

?>

Oya... satu lagi, setiap akan menampilkan nilai session perintah `session_start()` harus diberikan terlebih dahulu. Perhatikan pada script di atas untuk melihat hal ini!

Setelah menggunakan session, dapat Anda lihat bahwa username ini akan selalu tampak pada setiap halaman yang ada.

Nah... mungkin ada pertanyaan lain. Bagaimana cara menghapus nilai session yang telah tersimpan? Nilai session ini akan terhapus otomatis begitu browser ditutup atau keluar dari browser. Cara lain adalah dengan menggunakan perintah `session_destroy()` atau `unset($_SESSION['nama session'])`. Bedanya apa antara kedua perintah tersebut? `session_destroy()` digunakan untuk menghapus semua session. Jadi, misalkan dalam script Anda terdapat 10 nama session, dan misalkan Anda ingin menghapus semua session tersebut maka gunakan `session_destroy()`. Lalu `unset($_SESSION['nama session'])` digunakan untuk menghapus session tertentu saja.

Konsep penghapusan nilai session ini, dapat diterapkan pada proses logout. Karena pada prinsipnya proses logout ini adalah menghapus nilai session (dalam hal ini adalah username) yang telah tersimpan. Berikut ini contoh script logout.

### **logout.php**

```
<?php  
session_start();  
  
unset($_SESSION['namauuser']);  
echo "Anda telah logout";  
?>
```

Apabila script logout ini dijalankan, maka username yang telah tersimpan tadi tidak akan muncul lagi di halaman-halaman yang ada.

Manfaat session yang lain adalah dapat mencegah user mengakses halaman-halaman tertentu yang sifatnya private tanpa melakukan login (by pass). Dalam contoh di atas, Anda akan dapat mengakses halaman 1, halaman 2 dan 3 secara langsung tanpa proses login terlebih dahulu. Nah... dengan session, Anda dapat membuat ketiga halaman tersebut tidak bisa diakses oleh user yang masuk tanpa proses login. Idenya adalah dengan mendeteksi session username. Pendekstian ini dilakukan di ketiga halaman tersebut. Bila terdeteksi nilai session username ini masih kosong, maka dianggap user yang mengakses tersebut tidak melakukan login terlebih dahulu, sehingga akses harus diblok. Berikut ini script untuk mendeteksi session username yang masih kosong.

### **cek.php**

```
<?php
```

```

session_start();

if (!isset($_SESSION['namauser']))
{
    echo "Anda belum login";
    exit;
}

?>

```

Script di atas nantinya akan disisipkan ke ketiga halaman private menggunakan include(). Perintah ini disisipkan sebelum menampilkan konten yang ada pada halaman tersebut. Berikut ini contoh menyisipkan script cek.php ke halaman pertama. Untuk halaman yang lain, caranya sama.

### hal1.php

```

<?php
session_start();
include "cek.php";

echo "<h1>Ini halaman pertama</h1>";
echo "<p>Anda login sebagai ".$_SESSION['namauser']."</p>";
echo "<p>Berikut ini menu navigasi Anda</p>";
echo "<p><a href='hal1.php'>Menu 1</a> <a href='hal2.php'>Menu 2</a> <a href='hal3.php'>Menu 3</a></p>";
?>

```

Begitu terdeteksi user yang mengakses halaman tersebut tanpa login, maka akan muncul ‘Anda belum login’, sedangkan konten halaman aslinya tidak akan muncul. Hal ini karena efek dari perintah ‘exit’. Penjelasan ini pernah saya tulis di artikel [Membuat Autentifikasi User dengan PHP](#)<sup>[1]</sup>.

OK... demikian penjelasan penggunaan session dan manfaatnya. Mudah-mudahan penjelasan panjang lebar ini bisa bermanfaat bagi Anda semuanya.

## 57. Script PHP untuk Membuat Dokumen RTF

Artikel ini terinspirasi oleh requestnya mas “Kereayem” yang menanyakan bagaimana cara membuat dokumen DOC dengan script PHP di [salah satu komentarnya dalam blog ini](#)<sup>[1]</sup>. Namun maaf mas... karena sepertinya dokumen DOC sulit dimanipulasi dengan PHP, sehingga sebagai gantinya adalah dokumen dalam bentuk RTF. Setelah cari referensi sana sini terkait dengan hal ini, akhirnya ketemu juga ilmunya. Nah... sekarang saatnya saya berbagi ilmu tersebut melalui blog ini.

OK... dalam artikel ini akan dijelaskan bagaimana cara mengenerate dokumen RTF (Rich Text Format) dengan script PHP. Mengapa RTF? kenapa bukan DOC nya MS. Word? Yup... good question. Konten dokumen RTF lebih mudah untuk dimanipulasi dibandingkan dokumen DOC. Hal ini dikarenakan dalam dokumen DOC terdapat unsur grafis, dan bukanlah text based. Untuk menunjukkan hal ini, cobalah dokumen DOC ini Anda buka melalui text editor seperti NOTEPAD, atau Crimson Editor dll, pastilah isinya semrawut dan di sana sini tampil kode-kode yang tidak bisa dibaca. Namun beda halnya bila Anda membuka file RTF dengan text editor, pastilah isinya tidak semrawut dan dapat dibaca. Ya... dokumen RTF merupakan dokumen berbasis text based. Oleh karena itu, isi dokumen RTF ini dapat kita manipulasi dengan script untuk mendapatkan dokumen output yang diinginkan.

Dalam contoh ini, studi kasus yang diambil adalah membuat surat undangan dimana nama orang yang diundang, alamatnya, serta tempat dan waktu undangan akan diinputkan melalui suatu form. Setelah form ini diproses, selanjutnya akan muncul surat undangan berbentuk dokumen RTF dengan isi sesuai dengan data yang dimasukkan via form tadi.

Untuk membuat studi kasus di atas, apa yang dilakukan pertama kali? Hal pertama yang Anda lakukan adalah membuat template dokumen untuk surat undangannya. Caranya adalah bukalah MS. Word, lalu ketikkan sebarang format surat undangan, misalnya seperti di bawah ini:

Hal : Undangan

Kepada Ykh.  
Bp/Ibu %%NAMA%%  
Di %%ALAMAT%%

Dengan hormat,  
Mengharap kehadiran Bapak/Ibu untuk menghadiri rapat koordinasi  
yang akan diadakan pada:

Tanggal : %%TGL%%  
Tempat : %%TEMPAT%%  
Waktu : %%WAKTU%%

Demikian undangan ini dibuat, atas kehadirannya disampaikan  
terimakasih.

Surakarta, 5 Nopember 2008

Sekretaris  
Rosihan Ari Yuana

Perhatikan template di atas! Dalam template terdapat penanda untuk masing-masing item, misalnya untuk item nama yang diundang diberi tanda %%NAMA%%, tanggal undangannya diberi tanda %%TGL%% dsb. Tanda-tanda inilah yang nantinya akan digantikan dengan data yang disubmit via form. Apakah tandanya harus diapit dengan %% ? Sebaiknya ya.. karena tanda % dalam RTF tetap ditulis apa adanya. Maksudnya, bila Anda menulis % di dokumen RTF, lalu membukanya di text editor maka tanda % tadi tidak akan berubah. Namun beda hasilnya bila Anda menggunakan tanda #, misalnya. Bila Anda ketika # di dokumen, lalu membukanya di text editor, maka tanda ini akan dinyatakan dalam bentuk yang lain.

Setelah Anda buat dokumen templatenya, sekarang Anda simpan template tersebut ke dalam format RTF. Caranya klik menu FILE di MS. Word, pilih SAVE, pada bagian SAVE AS TYPE, pilih (Rich Text Format) RTF. Misalkan dokumen tersebut kita simpan dengan nama file surat.rtf.

Nah... untuk iseng-iseng saja, sekarang coba bukalah surat.rtf ini dengan NOTEPAD, NOTEPAD++, Crimson Editor atau text editor yang lain. Pastilah isi yang tampil adalah sebagai berikut:

(dipotong, karena panjang....)

```
\fs24\lang1033\langfe1033\cgrid\langnp1033\langfenp1033
{\insrsid1652669 Hal\tab : Undangan
\par
\par Kepada Ykh.
\par Bp/Ibu %%NAMA%%
\par Di %%ALAMAT%%
\par
\par Dengan hormat,
\par Mengharap kehadiran Bapak/Ibu untuk menghadiri rapat
koordinasi yang akan diadakan pada:
\par
\par Tanggal\tab : %%TGL%%
\par Tempat\tab : %%TEMPAT%%
\par Waktu\tab \tab : %%WAKTU%%
\par
\par Demikian undangan ini dibuat, atas kehadirannya disampaikan
terimakasih.
\par
\par Surakarta, 5 Nopember 2008
\par
\par Sekretaris
\par
\par
\par Rosihan Ari Yuana
\par }}
```

Catatan: tempate di atas dibuat dengan MS. Word 2003. Bisa jadi isi dokumen RTF di atas berbeda untuk setiap versi MS. Word.

Anda perhatikan isi template, bahwa tanda-tanda untuk setiap item yang kita buat tadi masih terlihat. Sehingga tugas kita selanjutnya adalah melakukan replacing string untuk masing-masing penanda dengan data yang diisikan via form. Itu adalah ide dasarnya.

OK... template surat sudah dibuat, langkah selanjutnya adalah membuat form untuk mengisi data surat. Ini dia contoh formnya.

```
<h1>Buat Surat Undangan</h1>
1 <form method="post" action="surat.php">
2 <table>
3 <tr><td>Nama Penerima Surat</td><td><input type="text"
4 name="nama"></td></tr>
5 <tr><td>Alamat Penerima Surat</td><td><input type="text"
6 name="alamat"></td></tr>
7 <tr><td>Tanggal Acara</td><td><input type="text" name="tanggal"></td></tr>
8 <tr><td>Tempat Acara</td><td><input type="text" name="tempat"></td></tr>
9 <tr><td>Waktu Acara</td><td><input type="text" name="waktu"></td></tr>
10 <tr><td></td><td><input type="submit" name="submit"
11 value="Submit"></td></tr>
12 </table>
</form>
```

Berikutnya, kita buat script surat.php untuk memproses form. Di dalam script PHP ini pastilah terdapat perintah untuk melakukan replacing string masing-masing penanda dengan data string dari form. Apa perintah yang bisa digunakan? Untuk replacing ini kita bisa gunakan str\_replace().

surat.php

```
1 <?php
2
3 // membaca data dari form
4
5 $nama = $_POST['nama'];
6 $alamat = $_POST['alamat'];
7 $stanggal = $_POST['tanggal'];
8 $waktu = $_POST['waktu'];
9 $tempat = $_POST['tempat'];
10
11 // membaca isi dokumen tempate surat.rtf
12 // isi dokumen dinyatakan dalam bentuk string
13
14 $document = file_get_contents("surat.rtf");
15
16 // mereplace tanda %%NAMA% dengan data nama dari form
17 $document = str_replace("%%NAMA%%", $nama, $document);
18
19 // mereplace tanda %%ALAMAT% dengan data alamat dari form, dst
20 $document = str_replace("%%ALAMAT%%", $alamat, $document);
21
22 $document = str_replace("%%TGL%%", $stanggal, $document);
23 $document = str_replace("%%TEMPAT%%", $tempat, $document);
```

```
24 $document = str_replace("%%WAKTU%%", $waktu, $document);
25
26 // header untuk membuka file output RTF dengan MS. Word
27 // nama file output adalah undangan.rtf
28
29 header("Content-type: application/msword");
30 header("Content-disposition: inline; filename=undangan.rtf");
31 header("Content-length: " . strlen($document));
32 echo $document;
33
34 ?>
```

Begitulah cara untuk mengenerate dokumen RTF dengan script PHP. Memang dalam contoh ini data yang disisipkan ke dokumen RTF bersifat single data. Untuk multiple data, seperti misalnya menampilkan data-data hasil query ke dalam bentuk tabel pada dokumen RTF, hal ini agak sulit dilakukan. Hal ini dikarenakan format tabel dalam dokumen RTF sangat kompleks bentuknya. Tapi bukan berarti tidak mungkin dibuat. Bila ada di antara Anda yang sudah berhasil membuat dokumen RTF dengan multiple data ini, dapat dishare ilmunya ke saya



OK... selamat mencoba

NB:

Anda dapat mendownload script di atas (termasuk dokumen RTF template suratnya), di bawah ini

[\*\*Download Script\*\*](#) [2]

## 58. Menguak Rahasia Penggunaan Cookies di Bisnis Afiliasi

Tentu Anda sudah pernah mendengar istilah bisnis afiliasi atau affiliate business, khususnya bisnis afiliasi melalui dunia maya atau internet. Biasanya orang yang ikut ikut bisnis ini, akan diberikan suatu link unik dengan bentuk <http://namasitus/?id=...> atau <http://namasitus/?ref=...>

Karakteristik lain dari bisnis afiliasi ini adalah, mengiming-imingi para calon konsumennya dengan lamanya cookies, misalnya ‘Bisnis afiliasi X memiliki cookies selama 10 tahun, 2 tahun, bahkan ada yang cuma dalam satuan bulan’.

Nah apa maksudnya suatu bisnis afiliasi memiliki cookies selama 10 tahun, dan apa pula hubungan antara link unik dengan cookies ini? Bagaimana pula cara membuat cookies untuk bisnis afiliasi ini dengan PHP? Semuanya ini akan terkuak dalam artikel ini.

OK... kita mulai dahulu pembahasan ini dari link unik yang diberikan situs bisnis afiliasi kepada setiap membernya. Setiap member bisnis afiliasi pasti memiliki id unik. Nah id ini nanti akan disisipkan pada setiap link unik. Fungsi link unik berisi id ini supaya script afiliasi bisa mendeteksi id member siapakah yang akan mendapat komisi, bila ada orang lain yang membeli produk atau ikut gabung dalam bisnis ini melalui link member.

Sebagai contoh, misalnya ada seseorang bernama Rosihan Ari memiliki id 123. Orang ini mendapatkan link unik <http://namasitus/?ref=123>. Ketika Rosihan Ari ini menaruh link tersebut di situs pribadinya, atau mengirimnya lewat email dan ternyata ada orang lain atau temannya yang mengklik link tersebut, maka Rosihan Ari ini akan mendapat komisi atau bagi hasil atau bonus dari bisnis afiliasi ini, dengan catatan orang lain tersebut harus telah membeli produk atau ikut bergabung ke dalam bisnis, tergantung ketentuan dari bisnis afiliasi ini.

Trus... apa maksudnya bahwa bisnis afiliasi menawarkan cookies dengan lama tertentu, misalnya 2 tahun. Nah... hal ini maksudnya, jika ada orang lain yang mengklik link uniknya Rosihan Ari, maka cookies berisi id miliknya Rosihan Ari ini akan tersimpan di dalam komputer orang yang mengklik tersebut, dan tersimpan selama 2 tahun.

Lalu.... fungsinya apa tuh cookies? OK... untuk menjelaskannya, saya akan ambil contoh kasus. Misalnya orang lain ini mengklik link unik miliknya Rosihan Ari pada tanggal 1 Januari 2008. Namun, orang tadi tidak jadi memutuskan untuk membeli produk yang ditawarkan bisnis afiliasi pada saat itu. Tapi suatu saat, dalam jangka waktu kurang atau sama dengan 2 tahun (misalnya tanggal 1 Januari 2009) ternyata orang tersebut memutuskan untuk membeli produk di situs bisnis afiliasi tersebut, maka si Rosihan Ari ini tetap akan mendapat komisi. Lho kok bisa? Mengapa demikian, ya... karena cookies yang berisi id nya Rosihan Ari ini tersimpan di komputernya orang lain tadi. Dengan catatan, komputer yang digunakan orang lain tersebut ketika mengklik link unik pertama kali (tgl 1 Januari 2008) dengan ketika membeli produk (tgl 1 Januari 2009) adalah sama, atau selama cookiesnya belum terhapus.

Oleh karena itu, bisnis afiliasi ini akan menjadi lebih menarik dan diminati oleh banyak orang bila semakin lama masa aktif cookiesnya. Bahkan ada pula bisnis afiliasi yang

menawarkan lifetime cookies, artinya masa aktif cookies adalah seumur hidup. Wow... Akan tetapi bila suatu bisnis afiliasi yang sudah kondang, dan banyak membernya biasanya menggunakan cookies dalam satuan bulan saja, misalnya 3 bulan atau bahkan 1 bulan.

Nah selanjutnya, bagaimana cara membuat script PHP untuk proses penyimpanan cookies ini di komputer, kemudian bagaimana pula membuat script untuk membaca cookies dan mendeteksi id membernya?

Selanjutnya dalam artikel bagaimana cara membuat script PHP untuk menyimpan cookies dan membaca cookies supaya mendeteksi id member yang telah memberikan referensi kepada orang lain untuk membeli produk bisnis afiliasi. Orang yang mereferensikan ini biasa disebut Referral.

Untuk mengingat kembali fungsi link unik referral, saya ambil contoh misalkan ada seorang member bisnis afiliasi bernama Rosihan Ari. Dia diberikan link unik <http://namasitus/?ref=123> (no. id Rosihan Ari adalah 123). Trus... Rosihan Ari ini menaruh link tersebut di situs pribadinya, dan kebetulan ada orang misalkan bernama X yang mengklik link tersebut.

Begitu link tersebut diklik, secara otomatis cookies berisi id Rosihan Ari akan tersimpan di dalam komputernya si X tadi.

Bagaimana script PHP nya? Ini dia rahasianya.

```
1 $lamaCookies = 2; // lama cookies 2 tahun
2 if (isset($_GET['ref']))
3 {
4     $_SESSION['ref'] = $_GET['ref'];
5     setcookie("ref", $_SESSION['ref'], time() +3600*24*365*$lamaCookies);
6 }
```

Baris pertama dari script di atas, merupakan setting untuk mengatur lama usia cookies (dalam contoh ini adalah 2 tahun). Baris kedua digunakan untuk mengecek, apakah orang lain ketika masuk ke situs bisnis afiliasi melalui link unik membernya? di sini akan dicek apakah terdapat embel-embel ?ref=... atau tidak. Artinya kalau ada seseorang masuk ke situs bisnis afiliasi tanpa melalui link unik salah satu membernya (<http://namasitus>), maka nilai `isset($_GET['ref'])` bernilai false, sehingga perintah di dalam IF tidak dijalankan.

Bagaimana bila ada orang masuk ke situs melalui link unik member (<http://namasitus/?ref=...>)? Bila demikian, maka nilai `isset($_GET['ref'])` bernilai TRUE. Sehingga masuk ke bagian dalam IF.

Nah... ada apa dalam IF? Di dalam IF, terdapat perintah untuk menyimpan cookies. Baris ke-4, digunakan untuk menyimpan id referral ke dalam session. Apa gunanya? session ini berguna untuk mencatat id referral selama kunjungan seseorang di situs. Jika orang tersebut melakukan transaksi atau pembelian produk, maka id referral yang tercatat ini nanti akan

mendapatkan bonus. Selain itu session berisi id referal ini juga akan disimpan ke dalam cookies, yang dalam hal ini dilakukan pada baris ke-5 dari script di atas.

Coba perhatikan perintah setcookie() tersebut. Terdapat 3 parameter pada setcookie(). Parameter pertama digunakan untuk memberi nama cookie. Parameter kedua digunakan untuk menyatakan nilai yang disimpan dalam cookie, dalam hal ini adalah id referal. Sedangkan parameter ketiga untuk setting lama usia cookie, dalam satuan detik. Function time() digunakan untuk mencatat timestamp waktu sekarang. Bila usia cookie yang tersimpan dalam komputer sudah habis, maka secara otomatis cookie ini akan terhapus.

Selanjutnya akan kita bahas bagaimana script untuk membaca id referal yang berasal dari cookie yang tersimpan. Ini dia rahasianya.

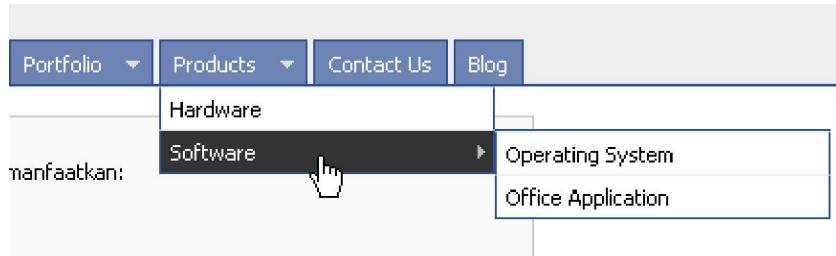
```
1 $ref = $_SESSION['ref'];
2
3 if (empty($ref))
4 {
5     $ref = $_COOKIE['ref'];
6     if (empty($ref))
7     {
8         $ref = 0;
9     }
10 }
```

Baris pertama pada script di atas digunakan untuk membaca session berisi id referal. Perintah ini berguna bila ada seseorang yang langsung memutuskan untuk membeli produk di bawah id referal seseorang pada saat itu juga, dalam hal ini, nilai \$\_SESSION['ref'] akan berisi id referal. Namun... bila orang tersebut memutuskan untuk menunda pembelian, dalam hal ini nilai \$\_SESSION['ref'] nya kosong, maka script akan memeriksa cookie. Bila cookienya ada, maka script akan membaca id referal dalam cookie (menggunakan perintah \$\_COOKIE['ref']). Tetapi... jika cookienya kosong, maka dianggap tidak ada referal sama sekali.

Mungkin Anda bertanya, dimanakah kedua script di atas diletakkan. Script-script tersebut diletakkan di halaman index atau di bagian header pada situs bisnis.

OK... demikian penjelasan tentang bagaimana membuat script untuk meletakkan cookie dan membacanya yang terkait dengan bisnis afiliasi. Mudah-mudahan berguna bagi para pengembang situs bisnis afiliasi. Maju terus bisnis afiliasi di Indonesia.

## 59. Menu Horizontal Multi Level Menggunakan jQuery & PHP



Di dalam sebuah website, seringkali kita menemukan penggunaan menu bertingkat seperti pada gambar di atas. Menu yang bertingkat tersebut sebenarnya adalah sebuah struktur pohon/tree di mana suatu menu bisa memiliki anak (submenu), dan submenu tersebut juga bisa memiliki anak lagi dan seterusnya sampai level tak terbatas.

Berikut ini adalah contoh struktur html menu yang terdiri dari 3 level (menggunakan tag html `<ul>` dan `<li>`):

```
1 <ul>
2   <li><a href="menu1.html">Menu 1</a></li>
3   <li><a href="menu2.html">Menu 2</a></li>
4   <li><a href="menu3.html">Menu 3</a>
5     <ul>
6       <li><a href="menu31.html">Sub Menu 3.1</a></li>
7       <li><a href="menu32.html">Sub Menu 3.2</a>
8         <ul>
9           <li><a href="menu321.html">Sub Menu 3.2.1</a></li>
10          <li><a href="menu322.html">Sub Menu 3.2.2</a></li>
11          <li><a href="menu323.html">Sub Menu 3.2.3</a></li>
12        </ul>
13      </li>
14      <li><a href="menu33.html">Sub Menu 3.3</a></li>
15    </ul>
16  </li>
17  <li><a href="menu4.html">Menu 4</a></li>
18 </ul>
```

Kode html tersebut akan menghasilkan tampilan seperti ini:

- [Menu 1](#)
- [Menu 2](#)
- [Menu 3](#)
  - [Sub Menu 3.1](#)
  - [Sub Menu 3.2](#)
    - [Sub Menu 3.2.1](#)
    - [Sub Menu 3.2.2](#)
    - [Sub Menu 3.2.3](#)
  - [Sub Menu 3.3](#)
- [Menu 4](#)

Dengan menggunakan jquery dan plugin [jquerycssmenu](#), menu tersebut bisa kita ubah menjadi sebuah menu dropdown yang lebih menghemat tempat. Tinggal tambahkan script berikut ini di dalam head:

```
1 <link rel="stylesheet" type="text/css" href="jquerycssmenu.css" />
2
3 <script type="text/javascript" src="jquery-1.3.2.min.js"></script>
4 <script type="text/javascript" src="jquerycssmenu.js"></script>
```

### Membuat menu dinamis dari database

Untuk membuat menu secara dinamis, kita bisa menggunakan database untuk menyimpan data menunya. Di bawah ini adalah struktur tabel untuk menyimpan menu:

```
1 CREATE TABLE `menu` (
2   `id` tinyint(3) UNSIGNED NOT NULL AUTO_INCREMENT,
3   `parent_id` tinyint(3) UNSIGNED NOT NULL DEFAULT '0',
4   `title` varchar(100) NOT NULL DEFAULT '',
5   `url` varchar(100) NOT NULL DEFAULT '',
6   `menu_order` tinyint(3) UNSIGNED NOT NULL DEFAULT '0',
7   PRIMARY KEY (`id`)
8 ) ENGINE=MyISAM;
```

Kemudian gunakan script php ini untuk membuat list secara dinamis dari database tersebut. Script ini menggunakan sebuah fungsi rekursif untuk mengambil menu sampai ke level tak terbatas.

```
1 <?php
2 function get_menu($data, $parent = 0) {
3     static $i = 1;
4     $tab = str_repeat("\t\t", $i);
5     if (isset($data[$parent])) {
6         $html = "\n$tab<ul>";
7         $i++;
8         foreach ($data[$parent] as $v) {
9             $child = get_menu($data, $v->id);
10            $html .= "\n\t$tab<li>";
11            $html .= '<a href="'.$v->url.'">'.$v->title.'</a>';
12            if ($child) {
13                $i--;
14                $html .= $child;
15                $html .= "\n\t$tab";
16            }
17            $html .= '</li>';
18        }
19        $html .= "\n$tab</ul>";
20        return $html;
21    } else {
22        return false;
23    }
24 }
25
26 mysql_connect('localhost', 'root', '');
```

```
27 mysql_select_db('tes');  
28  
29 $result = mysql_query("SELECT * FROM menu ORDER BY menu_order");  
30 while ($row = mysql_fetch_object($result)) {  
31     $data[$row->parent_id][] = $row;  
32 }  
33  
34 $menu = get_menu($data);  
35 ?>
```

[Lihat demonya di sini](#)

Download script ini:



## 60. jQuery Mega Menu

Mega Menus refer to drop down menus that contain multiple columns of links. This jQuery script lets you add a mega menu to any anchor link on your page, with each menu revealed using a sleek expanding animation. Customize the animation duration plus delay before menu disappears when the mouse rolls out of the anchor. Mega cool!

### Directions

**Step 1:** Add the following code to the <head> section of your page:

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/jquery.min.js"></script>

<link rel="stylesheet" type="text/css" href="jkmegamenu.css" />

<script type="text/javascript" src="jkmegamenu.js">

/*
 * jQuery Mega Menu- by JavaScript Kit (www.javascriptkit.com)
 * This notice must stay intact for usage
 * Visit JavaScript Kit at http://www.javascriptkit.com/ for full source code
 */

</script>

<script type="text/javascript">

//jkmegamenu.definemenu("anchorid", "menuid", "mouseover|click")
jkmegamenu.definemenu("megaanchor", "megamenu1", "mouseover")

</script>
```

It references two external files. Download them below (right click, and select "save as"):

- [jkmegamenu.css](#)
- [jkmegamenu.js](#)

**Step 2:** Add the below HTML code to your page. It includes one example anchor link plus the associated mega drop down menu:

```
<!--Mega Menu Anchor-->
<a href="http://www.javascriptkit.com" id="megaanchor">Tech Links</a>
<!--Mega Drop Down Menu HTML. Retain given CSS classes-->
<div id="megamenu1" class="megamenu">
```

```

<div class="column">
    <h3>Web Development</h3>
    <ul>
        <li><a href="http://www.javascriptkit.com">JavaScript Kit</a></li>
        <li><a href="http://www.dynamicdrive.com/">Dynamic Drive</a></li>
        <li><a href="http://www.cssdrive.com">CSS Drive</a></li>
        <li><a href="http://www.codingforums.com">Coding Forums</a></li>
        <li><a href="http://www.javascriptkit.com/domref/">DOM Reference</a></li>
    </ul>
</div>

<div class="column">
    <h3>News Related</h3>
    <ul>
        <li><a href="http://www.cnn.com/">CNN</a></li>
        <li><a href="http://www.msnbc.com">MSNBC</a></li>
        <li><a href="http://www.google.com">Google</a></li>
        <li><a href="http://news.bbc.co.uk">BBC News</a></li>
    </ul>
</div>

<div class="column">
    <h3>Technology</h3>
    <ul>
        <li><a href="http://www.news.com/">News.com</a></li>
        <li><a href="http://www.slashdot.com">SlashDot</a></li>
        <li><a href="http://www.digg.com">Digg</a></li>
        <li><a href="http://www.techcrunch.com">Tech Crunch</a></li>
    </ul>
</div>

<br style="clear: left" /> <!--Break after 3rd column. Move this if desired-->

<div class="column">
    <h3>Web Development</h3>
    <ul>
        <li><a href="http://www.javascriptkit.com">JavaScript Kit</a></li>
        <li><a href="http://www.dynamicdrive.com/">Dynamic Drive</a></li>
        <li><a href="http://www.cssdrive.com">CSS Drive</a></li>
        <li><a href="http://www.codingforums.com">Coding Forums</a></li>
        <li><a href="http://www.javascriptkit.com/domref/">DOM Reference</a></li>
    </ul>
</div>

<div class="column">
    <h3>News Related</h3>
    <ul>
        <li><a href="http://www.cnn.com/">CNN</a></li>
        <li><a href="http://www.msnbc.com">MSNBC</a></li>
        <li><a href="http://www.google.com">Google</a></li>
        <li><a href="http://news.bbc.co.uk">BBC News</a></li>
    </ul>
</div>

```

```

<div class="column">
    <h3>Technology</h3>
    <ul>
        <li><a href="http://www.news.com">News.com</a></li>
        <li><a href="http://www.slashdot.com">SlashDot</a></li>
        <li><a href="http://www.digg.com">Digg</a></li>
        <li><a href="http://www.techcrunch.com">Tech Crunch</a></li>
    </ul>
</div>
</div>

```

### Configuration info

Each Mega Menu consists of an HTML anchor link, plus its associated drop down menu. The anchor link should just be any arbitrary link with a unique ID attribute:

```
<!--Mega Menu Anchor-->
<a href="http://www.javascriptkit.com" id="megaanchor">Tech Sites</a>
```

The associated drop down menu on the other hand should be a DIV containing a series of ULs in the format shown in the code of Step 2. It should also carry a unique ID:

```
<!--Mega Drop Down Menu HTML. Retain given CSS classes-->
<div id="megamenu1" class="megamenu">
Mega Menu HTML here...
</div>
```

With both of the above components defined on your page, in the HEAD section of your page, initialize this menu by calling:

```
<script type="text/javascript">

//jkmegamenu.definemenu("anchorid", "menuid", "mouseover|click")
jkmegamenu.definemenu("megaanchor", "megamenu1", "mouseover")

</script>
```

Where the first two parameters are the IDs of the anchor and associated drop down menu, and the 3rd, a string of either "mouseover" or "click". This last parameter lets you specify which of these two actions the menu should be activated on the anchor link.

Finally, yes, you can have multiple mega menus on your page. Just repeat the steps above and call `jkmegamenu.definemenu()` for each menu. Enjoy!

## 61. Export hasil query MySql di web ke Excel

```
<form id="form1" name="form1" method="post" action="excel.php">

<p>
QUERY : <br />

<textarea name="sql" cols="50" rows="6" id="sql"></textarea>

</p>

<p>
<input type="submit" name="Submit" value="Submit" />

</p>

</form>

<p style="text-align: justify;"><strong>Nama File : excel.php</strong>
<?php

$hostname = "localhost";
$username = "root";
$database = "a";
$password = "";
$conn = mysql_connect($hostname,$username,$password);
mysql_select_db($database);

if (isset($_POST['sql']))
{
$q = mysql_query(stripslashes($_POST['sql'])) or die ("Not valid query !");
$res = "";
while ($f = mysql_fetch_field($q)) {
$res .= $f->name . "\t";
}
header("Content-type: application/vnd.ms-excel");
header("Content-Disposition: attachment; filename=hasil.xls");
echo $res;
}
else
{
header("Content-type: application/vnd.ms-excel");
header("Content-Disposition: attachment; filename=hasil.xls");
echo "No Query";
}
?>
```

```
}

$res .= "\r\n";

while ($r = mysql_fetch_array($q)) {

for ($i=0; $i<mysql_num_fields($q); $i++) {

$res .= $r[$i] . "\t";

}

$res .= "\r\n";

}

mysql_free_result($q);

header("Content-disposition: attachment; filename=aan.xls");

header("Content-type: Application/exe");

header("Content-Transfer-Encoding: binary");

echo $res;

}

?>
```

## 62. Menampilkan Format Angka ke Rupiah di PHP

```
//pencacah bilangan duit

function xduit($str) {
    //bernilai 3 digit

    if (strlen($str) == 3) {
        $nil1 = substr($str,-3);

        echo "$nil1";

    }

    //bernilai 4 digit

    else if (strlen($str) == 4) {
        $nil1 = substr($str,0,1);

        $nil2 = substr($str,-3);

        echo "$nil1.$nil2";

    }

    //jika ada 5 digit

    else if (strlen($str) == 5) {
        $nil1 = substr($str,0,2);

        $nil2 = substr($str,-3);

        echo "$nil1.$nil2";

    }

    //jika ada 6 digit

    else if (strlen($str) == 6) {
        $nil1 = substr($str,0,3);

        $nil2 = substr($str,-3);

        echo "$nil1.$nil2";
    }
}
```

```
}

//jika ada 7 digit

else if (strlen($str) == 7) {

$nil1 = substr($str,0,1);

$nil2 = substr($str,1,3);

$nil3 = substr($str,-3);

echo "$nil1.$nil2.$nil3";

}

//jika ada 8 digit

else if (strlen($str) == 8) {

$nil1 = substr($str,0,2);

$nil2 = substr($str,2,3);

$nil3 = substr($str,-3);

echo "$nil1.$nil2.$nil3";

}

//jika ada 9 digit

else if (strlen($harga) == 9) {

$nil1 = substr($str,0,3);

$nil2 = substr($str,3,3);

$nil3 = substr($str,-3);

echo "$nil1.$nil2.$nil3";

}

}
```

## 63. Shopping cart sederhana

Ketika anda memutuskan untuk membuat suatu usaha, salah satu marketingnya adalah dengan membuat website. Lewat website ini, produk-produk kita akan dikenal oleh orang banyak. Apalagi, dalam website tersebut terdapat sebuah aplikasi untuk shopping cart. Tentu sangat mudah sekali bagi customer untuk belanja online produk anda melalui website.

Berikut ini adalah contoh membuat shopping cart yang sangat sederhana. Contoh berikut ini, dapat anda kembangkan lagi sesuai kebutuhan anda.

Pertama, kita harus membuat database. Berilah nama database itu sesuai kebutuhan. Sebagai contoh nama databasenya adalah **butik**. Kemudian buatlah tabelnya dengan nama **produk**. Isi field2 dari tabel tersebut seperti contoh berikut :

localhost / localhost / butik / produk | phpMyAdmin 2.7.0-pl1 - ...

File Edit View History Bookmarks Tools Help

PMA http://localhost/php Google ABP

Most Visited Getting Started Latest Headlines

Butik

localhost / localhost / butik / ...

in horizontal mode and repeat  
headers after 100 cells

Sort by key: None Go

	id_produk	nama_produk	harga
<input type="checkbox"/>	1	baju	10000
<input type="checkbox"/>	2	celana	15000

Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat

Setelah selesai membuat database, sekarang saatnya ke bagian programmingnya. Programming yang digunakan adalah php.

Kedua, buatlah konfigurasi untuk koneksi ke database. Isinya :

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

kemudian save dengan nama **connect.php**

Ketiga, buat file dengan nama **index.php**. File ini berisi produk yang ada di database kita.

```
<html>
<head>

<title>Butik</title>
</head>

<body>
<form action="" method="post" enctype="multipart/form-data" name="form1" id="form1">
<table width="400" border="1" align="center">
<tr>
<td colspan="5" align="center">Pilih Salah Satu Daftar Produk</td>
</tr>
<tr>
<td align="center">No</td>
<td align="center">Nama Produk</td>
<td align="center">Harga</td>
<td align="center">Keterangan</td>
</tr>
<?php
include "connect.php"; // pastikan file ini ada
$sql = mysql_query("SELECT * FROM produk");
while ($row = mysql_fetch_array($sql))
{
    $id = $row['id_produk'];
    $nama = $row['nama_produk'];
    $hrg = $row['harga'];
?
<tr>
<td><?php echo $id; ?></td>
<td><?php echo $nama; ?></td>
<td>Rp. <?php echo $hrg; ?></td>
<td><a href="cart.php?product_id=<?php echo $id; ?>">Add to Cart</a></td>
</tr>
<?php
}
?
</table>
</form>
</body>
</html>
```



Keempat, buat file baru lagi dengan nama **cart.php**. File ini akan meminta inputan berapa barang yang ingin kita beli.

```
<html>
<head>

<title>Cart</title>
</head>

<body>
<?php
    include "connect.php";
    session_start();

    $id = (int) $_GET['product_id'];
    $qty = $_SESSION['qty'];

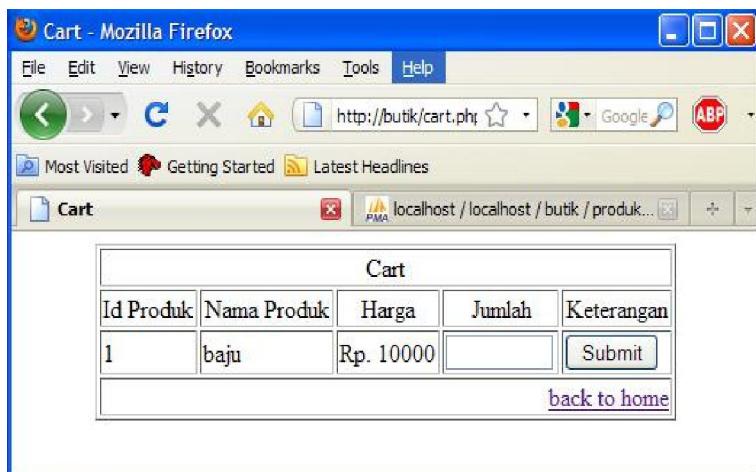
    $total = $_SESSION['total'];
?>
<form id="form1" name="form1" method="post" action="docart.php?product_id=<?php echo
$id; ?>">
    <?php
        if ($qty != "") {
            echo "Grand Total : $total<br>
                <a href=\"index.php\">back to home</a>";
        }
    else
    {
?>
```

```

<table width="400" border="1" align="center">
<tr>
<td colspan="5" align="center">Cart</td>
</tr>
<tr>
<td align="center">Id Produk</td>
<td align="center">Nama Produk</td>
<td align="center">Harga</td>
<td align="center">Jumlah</td>
<td align="center">Keterangan</td>
</tr>
<?php
$sql = mysql_query("SELECT * FROM produk WHERE id_produk='$id'");
if ($row = mysql_fetch_array($sql))
{
    $id = $row['id_produk'];
    $nama = $row['nama_produk'];
    $harga = $row['harga'];
?
<tr>
<td><?php echo $id; ?></td>
<td><?php echo $nama; ?></td>
<td>Rp. <?php echo $harga; ?></td>
<td><label><input type="text" name="qty" id="qty" size="8" value="<?php echo $qty; ?>"></label></td>
<td><label><input type="submit" name="button" id="button" value="Submit"></label></td>
</tr>
<tr>
<td colspan="5" align="right"><a href="index.php">back to home</a></td>
</tr>
<?php
}
}

session_destroy();
?>
</table>
</form>
</body>
</html>

```



Dan kelima, simpan file dibawah ini dengan nama docart.php. yang akan melakukan penghitungan dari jumlah yang diinput dengan harga yang ada.

```
<?php
    include "connect.php";
    session_start();

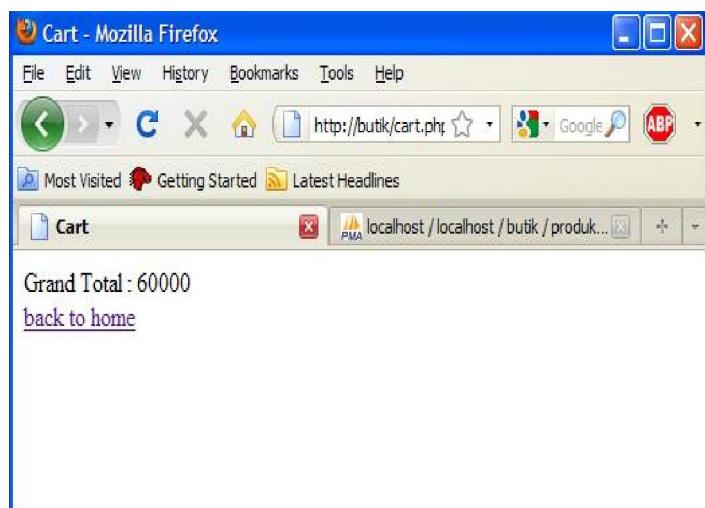
    $qty = $_REQUEST['qty'];
    $_SESSION['qty'] = $qty;

    $id_prod = (int) $_GET['product_id'];

    $sql = mysql_query("SELECT * FROM produk WHERE id_produk='$id_prod' ");
    if ($row = mysql_fetch_array($sql))
    {
        $hrg = $row['harga'];
        $_SESSION['harga'] = $hrg;
    }

    $total = $qty*$hrg;
    $_SESSION['total'] = $total;

    header("Location:cart.php");
?>
```



Biarpun cukup sederhana, tapi bisa anda gunakan untuk memenuhi kebutuhan website anda yang ingin menggunakan aplikasi shopping cart.

## 64. Creating a simple PHP forum tutorial

### Overview

In this tutorial create 5 files

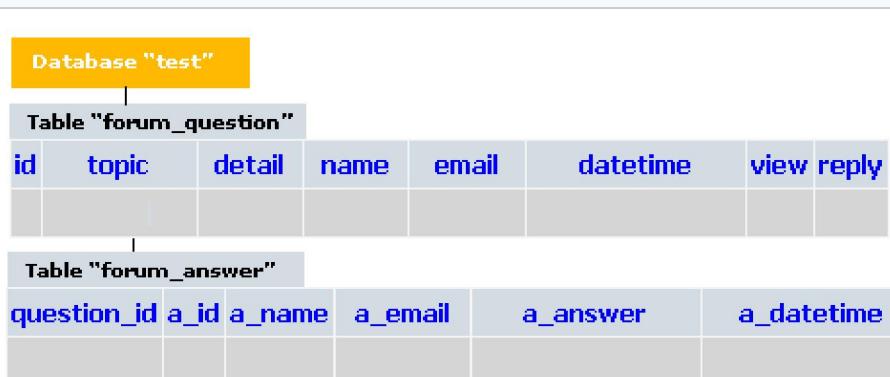
1. create\_topic.php
2. add\_topic.php
3. main\_forum.php
4. view\_topic.php
5. add\_answer.php

### Steps

1. Create table name "forum\_question" and "forum\_answer" in database "test".
2. Create file create\_topic.php.
3. Create file add\_topic.php.
4. Create file main\_forum.php
5. Create file view\_topic.php
6. Create file add\_answer.php

If you don't know how to create database and table [read this tutorial](#)

### 1 Set up database



#### Table forum\_question

CREATE TABLE `forum\_question` (

```

`id` int(4) NOT NULL auto_increment,
`topic` varchar(255) NOT NULL default '',
`detail` longtext NOT NULL,
`name` varchar(65) NOT NULL default '',
`email` varchar(65) NOT NULL default '',
`datetime` varchar(25) NOT NULL default '',
`view` int(4) NOT NULL default '0',
`reply` int(4) NOT NULL default '0',
PRIMARY KEY (`id`)
) TYPE=MyISAM AUTO_INCREMENT=1 ;

```

#### Table forum\_answer

```

CREATE TABLE `forum_answer` (
`question_id` int(4) NOT NULL default '0',
`a_id` int(4) NOT NULL default '0',
`a_name` varchar(65) NOT NULL default '',
`a_email` varchar(65) NOT NULL default '',
`a_answer` longtext NOT NULL,
`a_datetime` varchar(25) NOT NULL default '',
KEY `a_id` (`a_id`)
) TYPE=MyISAM;

```

#### 2 Create file create\_topic.php

[View in Browser](#)

Create New Topic

Topic :

Detail :

Name :

Email :

action="add\_topic.php"

## ##### Code

```
<table width="400" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#CCCCCC">
<tr>
<form id="form1" name="form1" method="post" action="add_topic.php">
<td>
<table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="#FFFFFF">
<tr>
<td colspan="3" bgcolor="#E6E6E6"><strong>Create New Topic</strong> </td>
</tr>
<tr>
<td width="14%"><strong>Topic</strong></td>
<td width="2%"></td>
<td width="84%"><input name="topic" type="text" id="topic" size="50" /></td>
</tr>
<tr>
<td valign="top"><strong>Detail</strong></td>
<td valign="top"></td>
<td><textarea name="detail" cols="50" rows="3" id="detail"></textarea></td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td></td>
<td><input name="name" type="text" id="name" size="50" /></td>
</tr>
<tr>
<td><strong>Email</strong></td>
<td></td>
<td><input name="email" type="text" id="email" size="50" /></td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td><input type="submit" name="Submit" value="Submit" /> <input type="reset" name="Submit2" value="Reset" /></td>
</tr>
</table>
</td>
</form>
</tr>
</table>
```

3

### Create file add\_topic.php

#### ##### Code

```
<?php
$host="localhost"; // Host name
$username=""; // Mysql username
$password=""; // Mysql password
$db_name="test"; // Database name
$tbl_name="forum_question"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// get data that sent from form
$topic=$_POST['topic'];
$detail=$_POST['detail'];
$name=$_POST['name'];
$email=$_POST['email'];

$datetime=date("d/m/y h:i:s"); //create date time

$sql="INSERT INTO $tbl_name(topic, detail, name, email, datetime)VALUES('$topic',
'$detail', '$name', '$email', '$datetime')";
$result=mysql_query($sql);

if($result){
echo "Successful<BR>";
echo "<a href=main_forum.php>View your topic</a>";
}
else {
echo "ERROR";
}
mysql_close();
?>
```

4

### Create file main\_forum.php

### View in dreamweaver / diagram

**View in Dreamweaver**

```

$host="localhost";
$username="";
$password="";
$db_name="test";
$tbl_name="forum_question";

mysql_connect("$host", "$username", "$password")
or die("cannot connect");
mysql_select_db("$db_name") or die("cannot select DB");

$sql="SELECT * FROM $tbl_name ORDER BY id DESC";
$result=mysql_query($sql);

```

A: Start looping table row

B: Exit looping

#	Topic	Views	Reply	Date/Time
<? echo \$rows['id']; ?>	<? echo \$rows['topic']; ?>	<? echo \$rows['view']; ?>	<? echo \$rows['reply']; ?>	<? echo \$rows['datetime']; ?>

Table "forum\_question"

<b>id</b>	<b>topic</b>	<b>detail</b>	<b>name</b>	<b>email</b>	<b>datetime</b>	<b>view</b>	<b>reply</b>

### ##### Code

```

<?php
$host="localhost"; // Host name
$username=""; // Mysql username
$password=""; // Mysql password
$db_name="test"; // Database name

```

```

$tbl_name="forum_question"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

$sql="SELECT * FROM $tbl_name ORDER BY id DESC";
// ORDER BY id DESC is order result by descending
$result=mysql_query($sql);
?>
<table width="90%" border="0" align="center" cellpadding="3" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td width="6%" align="center" bgcolor="#E6E6E6"><strong>#</strong></td>
<td width="53%" align="center" bgcolor="#E6E6E6"><strong>Topic</strong></td>
<td width="15%" align="center" bgcolor="#E6E6E6"><strong>Views</strong></td>
<td width="13%" align="center" bgcolor="#E6E6E6"><strong>Replies</strong></td>
<td width="13%" align="center" bgcolor="#E6E6E6"><strong>Date/Time</strong></td>
</tr>

<?php
while($rows=mysql_fetch_array($result)){ // Start looping table row
?>
<tr>
<td bgcolor="#FFFFFF"><? echo $rows['id']; ?></td>
<td bgcolor="#FFFFFF"><a href="view_topic.php?id=<? echo $rows['id']; ?>"><? echo
$rows['topic']; ?></a><BR></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['view']; ?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['reply']; ?></td>
<td align="center" bgcolor="#FFFFFF"><? echo $rows['datetime']; ?></td>
</tr>

<?php
// Exit looping and close connection
}
mysql_close();
?>
<tr>
<td colspan="5" align="right" bgcolor="#E6E6E6"><a
href="create_topic.php"><strong>Create New Topic</strong> </a></td>
</tr>
</table>

```

## 5 view\_topic.php

## View in Dreamweaver / Diagram

Topic table

php	
php	
php	
By : php	Email : php
Date/time : php	

answer table

ID	: php
Name	: php
Email	: php
Answer	: php
Date/Time	: php

php

Name	:	<input type="text"/>
Email	:	<input type="text"/>
Answer	:	<input type="text"/>  ↑ ↓
		<input type="button" value="Submit"/> <input type="button" value="Reset"/>

**A**

**View in Dreamweaver**

**From address bar**  
*/view\_topic.php?id=5*

```
<?php
$host="localhost";
$username="";
$password="";
$db_name="test";
$tbl_name="forum_question";
```

`mysql_connect("$host", "$username", "$password")
or die("cannot connect to server");
mysql_select_db("$db_name") or die("cannot select DB");`

`$sql="SELECT * FROM $tbl_name WHERE id='$id'";
$result=mysql_query($sql);`

`$rows=mysql_fetch_array($result);
?>`

**Table "forum\_question"**

<b>id</b>	<b>topic</b>

**topic table**

```
By : php Email : php <? echo $rows['topic']; ?>
Date/time : php <? echo $rows['detail']; ?>
<? echo $rows['name']; ?>
```

**Table "forum\_question"**

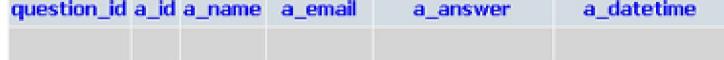
<b>id</b>	<b>topic</b>	<b>detail</b>	<b>name</b>	<b>email</b>	<b>datetime</b>	<b>view</b>	<b>reply</b>

B

```
<?php  
$tbl_name2="forum_answer";  
  
$sql2="SELECT * FROM $tbl_name2 WHERE question_id='$id"';  
$result2=mysql_query($sql2);  
  
while($rows=mysql_fetch_array($result2)){  
?>
```

PHP

ID	:  <? echo \$rows['a_id']; ?>	answer table
Name	:  <? echo \$rows['a_name']; ?>	
Email	:  <? echo \$rows['a_email']; ?>	
Answer	:  <? echo \$rows['a_answer']; ?>	
Date/Time	:  <? echo \$rows['a_datetime']; ?>	

 Table "forum\_answer"  


```
<?  
?>
```

```
$sql3="SELECT view FROM $tbl_name WHERE id='$id"';  
$result3=mysql_query($sql3);  
  
$rows=mysql_fetch_array($result3);  
$view=$rows['view'];  
  
// if have no view value set view = 1  
if(empty($view)){  
$view=1;  
$sql4="INSERT INTO $tbl_name(view) VALUES('$view') WHERE id='$id"';  
$result4=mysql_query($sql4);  
}  
  
// add more value  
$addview=$view+1;  
$sql5="update $tbl_name set view='$addview' WHERE id='$id"';  
$result5=mysql_query($sql5);  
  
mysql_close();  
?>
```

<input name="id" type="hidden" value="<? echo \$id; ?>">

## ##### Code

```

<?php
$host="localhost"; // Host name
$username=""; // Mysql username
$password=""; // Mysql password
$db_name="test"; // Database name
$tbl_name="forum_question"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// get value of id that sent from address bar
$id=$_GET['id'];

$sql="SELECT * FROM $tbl_name WHERE id='$id'";
$result=mysql_query($sql);

$rows=mysql_fetch_array($result);
?>


||
||
||


```

```

</tr>

<tr>
<td bgcolor="#F8F7F1"><strong>By :</strong> <? echo $rows['name']; ?> <strong>Email :</strong><? echo $rows['email']; ?></td>
</tr>

<tr>
<td bgcolor="#F8F7F1"><strong>Date/time :</strong><? echo $rows['datetime']; ?></td>
</tr>
</table></td>
</tr>
</table>
<BR>
<?php
$tbl_name2="forum_answer"; // Switch to table "forum_answer"

$sql2="SELECT * FROM $tbl_name2 WHERE question_id='$id'";
$result2=mysql_query($sql2);

while($rows=mysql_fetch_array($result2)){
?>
<table width="400" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#CCCCCC">
<tr>
<td><table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="#FFFFFF">
<tr>
<td bgcolor="#F8F7F1"><strong>ID</strong></td>
<td bgcolor="#F8F7F1">:</td>
<td bgcolor="#F8F7F1"><? echo $rows['a_id']; ?></td>
</tr>
<tr>
<td width="18%" bgcolor="#F8F7F1"><strong>Name</strong></td>
<td width="5%" bgcolor="#F8F7F1">:</td>
<td width="77%" bgcolor="#F8F7F1"><? echo $rows['a_name']; ?></td>
</tr>
<tr>
<td bgcolor="#F8F7F1"><strong>Email</strong></td>
<td bgcolor="#F8F7F1">:</td>
<td bgcolor="#F8F7F1"><? echo $rows['a_email']; ?></td>
</tr>
<tr>
<td bgcolor="#F8F7F1"><strong>Answer</strong></td>
<td bgcolor="#F8F7F1">:</td>
<td bgcolor="#F8F7F1"><? echo $rows['a_answer']; ?></td>
</tr>

```

```

<tr>
<td bgcolor="#F8F7F1"><strong>Date/Time</strong></td>
<td bgcolor="#F8F7F1">:</td>
<td bgcolor="#F8F7F1"><? echo $rows['a_datetime']; ?></td>
</tr>
</table></td>
</tr>
</table><br>

<?
}

$sql3="SELECT view FROM $tbl_name WHERE id='$id'";
$result3=mysql_query($sql3);

$rows=mysql_fetch_array($result3);
$view=$rows['view'];

// if have no counter value set counter = 1
if(empty($view)){
$view=1;
$sql4="INSERT INTO $tbl_name(view) VALUES('$view') WHERE id='$id'";
$result4=mysql_query($sql4);
}

// count more value
$addview=$view+1;
$sql5="update $tbl_name set view='$addview' WHERE id='$id'";
$result5=mysql_query($sql5);

mysql_close();
?>
<BR>
<table width="400" border="0" align="center" cellpadding="0" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<form name="form1" method="post" action="add_answer.php">
<td>
<table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="#FFFFFF">
<tr>
<td width="18%"><strong>Name</strong></td>
<td width="3%"></td>
<td width="79%"><input name="a_name" type="text" id="a_name" size="45"></td>
</tr>
<tr>
<td><strong>Email</strong></td>

```

```

<td></td>
<td><input name="a_email" type="text" id="a_email" size="45"></td>
</tr>
<tr>
<td valign="top"><strong>Answer</strong></td>
<td valign="top"></td>
<td><textarea name="a_answer" cols="45" rows="3" id="a_answer"></textarea></td>
</tr>
<tr>
<td>&nbsp;</td>
<td><input name="id" type="hidden" value=<? echo $id; ?>></td>
<td><input type="submit" name="Submit" value="Submit"> <input type="reset" name="Submit2" value="Reset"></td>
</tr>
</table>
</td>
</form>
</tr>
</table>

```

## **6 add\_answer.php**

##### Code

```

<?php
$host="localhost"; // Host name
$username=""; // Mysql username
$password=""; // Mysql password
$db_name="test"; // Database name
$tbl_name="forum_answer"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// Get value of id that sent from hidden field
$id=$_POST['id'];

// Find highest answer number.
$sql="SELECT MAX(a_id) AS Maxa_id FROM $tbl_name WHERE question_id='$id'";
$result=mysql_query($sql);
$rows=mysql_fetch_array($result);

// add + 1 to highest answer number and keep it in variable name "$Max_id". if there no answer
yet set it = 1
if ($rows) {
$Max_id = $rows['Maxa_id']+1;

```

```

}

else {
$Max_id = 1;
}

// get values that sent from form
$a_name=$_POST['a_name'];
$a_email=$_POST['a_email'];
$a_answer=$_POST['a_answer'];

$datetime=date("d/m/y H:i:s"); // create date and time

// Insert answer
$sql2="INSERT INTO $tbl_name(question_id, a_id, a_name, a_email, a_answer,
a_datetime)VALUES('$id', '$Max_id', '$a_name', '$a_email', '$a_answer', '$datetime')";
$result2=mysql_query($sql2);

if($result2){
echo "Successful<BR>";
echo "<a href='view_topic.php?id=".$id.">View your answer</a>";

// If added new answer, add value +1 in reply column
$tbl_name2="forum_question";
$sql3="UPDATE $tbl_name2 SET reply='$Max_id' WHERE id='$id'";
$result3=mysql_query($sql3);

}
else {
echo "ERROR";
}

mysql_close();
?>

```

## 65. Membuat Shopping Cart Sederhana Membuat Shopping Cart Sederhana Menggunakan PHP

### Pendahuluan

PHP adalah bahasa pemrograman web server-side yang sangat powerfull. PHP kini telah digunakan oleh ribuan website di seluruh dunia dan telah menghasilkan berbagai macam teknologi canggih, seperti blog, CMS, AJAX, dll. Bahasa PHP juga cocok untuk digunakan membuat situs toko online atau yang biasa disebut e-commerce.

Pada artikel ini saya akan menjelaskan cara membangun sebuah script Shopping Cart (keranjang belanja) sederhana. Script ini masih sangat sederhana dan bisa dikembangkan untuk digunakan di website anda masing-masing. Script ini juga hanya berupa Shopping Cart dan belum termasuk script untuk pembayaran atau manajemen produk.

### Cara kerja

Script ini bekerja dengan memanfaatkan variabel SESSION milik PHP dan database MySQL. Nantinya aplikasi ini berguna untuk menambah barang, meng-edit barang, dan membuang barang. Script ini nantinya dibagi menjadi 3 buah file yang berbeda yang bisa saling meng-include.

Struktur session

key	value
id_prod	jmlh_barang

Struktur Tabel 'produk'

Nama kolom	Tipe	Value	Tambahan
id_prod	int	10	auto increment, primary key
harga	int	20	
title	varchar	100	

Isi tabel 'produk'

id_prod	harga	title
1	1000	buku
2	100	permeh

### db.php

File ini akan berisi informasi untuk terhubung ke database MySQL kita nantinya. File ini sengaja dibuat terpisah agar memudahkan jika seandainya kita ingin memindahkan hosting script kita ini. Agar bisa terhubung, kita bisa cukup meng-include file ini di script yang kita butuhkan.

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

### view.php

File ini nantinya berguna untuk melihat jumlah barang yang ada di dalam shopping cart kita. Yang perlu kita perhatikan adalah jangan lupa meletakkan perintah `session_start()` di awal file. Anda bisa meng-include file ini ke setiap halaman yang ada. Berikut isi file tersebut:

```
<?php  
$total=0;  
if (!empty($_SESSION['basket'])) { //cek isi variabel _SESSION['basket']  
foreach ($_SESSION['basket'] as $key => $val) {  
$total +=$val; //menghitung jumlah barang  
}  
echo 'Ada <strong>' . $total . '</strong> barang di keranjang<br />';  
} else {  
echo "Tidak ada barang di keranjang";  
}  
?>
```

### cart.php

File inilah yang menjadi file utama kita. File inilah yang bertugas untuk menambah, mengubah, dan mengurangi isi shopping cart kita nantinya. Sekali lagi, jangan lupa untuk meletakkan perintah `session_start()` di awal file dan meng-include file db.php yng tadi kita buat.

```
<?php  
session_start(); //memulai session  
require_once "db.php"; //meng-include file db.php  
$pesan=''; //handler untuk aksi shopping cart  
if (isset($_GET['action'])) {  
if (isset($_GET['id'])) { //cek variabel id  
    $id=(int)$_GET['id'];  
} else {  
    $id=0;  
}  
$action=$_GET['action'];  
switch($_GET['action']) {  
case 'add':  
    if (!empty($_SESSION['basket'][$id])) {  
        $pesan="Barang sudah ada di keranjang";  
    } else {  
        $_SESSION['basket'][$id]=1;  
    }  
break;  
case 'update' :  
    $produk=$_POST['produk'];  
    foreach ($produk as $key => $val) {  
        if (!empty($_SESSION['basket'][$key])) {  
            $_SESSION['basket'][$key]=$val;  
        } //jika barang memang ada, baru di-update
```

```

        }
        break;
    case 'delete' :
        if (!empty($_SESSION['basket'][$id])) {
            unset($_SESSION['basket'][$id]);
        } else {
            $pesan="Barang yang dimaksud tidak ada";
        }
    break;
}
}

//Menampilkan isi shopping cart
if (!empty($_SESSION['basket'])) {
$basket=$_SESSION['basket'];
?>
<FORM method="POST" action="?action=update">
<h3>Isi keranjang belanja:</h3>
<center><table border="1">
<tr>
    <th>No.</th>
    <th>Nama produk</th>
    <th>Jumlah</th>
    <th>Aksi</th>
</tr>
<?php
$no_urut=0;
$total=0;
$total_harga=0;
foreach ($basket as $key => $val) { //menuliskan tabel
    $no_urut++;
    $query="SELECT * FROM produk WHERE id_prod='$key'";
    $hasil=mysql_query($query);
    if ($hasil) {
        $data=mysql_fetch_array($hasil);
        $total+=$val; //jumlah barang
        $total_harga += ($val * $data['harga']); //total harga
    }
    <tr>
        <td><?php echo $no_urut?></td>
        <td><?php echo $data['title']?></td>
        <td><INPUT name="produk[<?php echo $key?>]" type="text" value="<?php
echo $val?>" /></td>
        <td><A href="?action=delete&id=<?php echo $key?
">Hapus</a></A></td>
    </tr>
    <?php }
} ?>
<tr>
    <td colspan="2" align="center"><INPUT type="submit"
value="Perbaharui"></td>
    <td><strong>Total</strong></td>
    <td><?php echo 'Rp '. $total_harga?></td>
</tr>
</table>
</FORM><br />
<?php
echo 'Ada <strong>'. $total.'</strong> barang di keranjang<br />';

```

```
    } else {
echo "<center><strong>Tidak ada barang di
keranjang</strong><br />";
}
echo '<strong>'.$this->pesan.'</strong>'; //menampilkan pesan kalau ada
?></center>
```

### **Menambah, mengubah, atau membuang item dari shopping cart**

Untuk menambah item ke shopping cart, cukup sediakan link ke halaman yang meng-include file cart.php dengan tambahan url “?action=add&id=<id\_produk>”. Contoh, jika file yang menginclude file cart.php adalah toko.php dan no id tabel item yang akan ditambahkan adalah 1, maka sediakan link dengan target/tujuan: “toko.php?action=add&id=1”. Begitu diklik, maka halaman secara otomatis berganti ke halaman shopping cart.

Sementara untuk mengubah atau membuang item dari shopping cart, anda tidak perlu membuat link untuknya. Kedua hal di atas bisa dilakukan lewat halaman shopping cart. Anda hanya perlu memastikan bahwa ada link untuk menuju ke halaman shopping cart di setiap halaman situs webstore anda.

### **Penutup**

Script yang kita buat kali ini masih sangat sederhana. Masih banyak fitur yang belum di implementasikan di script ini, di antaranya script ini belum berupa class. Untuk mendapatkan script yang sudah berupa class, anda bisa mendapatkannya di alamat <http://blog.opentornado.com>. Anda bisa saja mengembangkannya menjadi sebuah script e-commerce yang lengkap. Jika anda sudah membuatnya, pastikan anda mengirim email ke saya agar saya bisa mengetahuinya.

Akhir kata, semoga artikel ini bisa bermanfaat bagi kita semua. Salam dan Semoga sukses!

***Abdurrahman Shofy Adianto***

[420phy@opentornado.com](mailto:420phy@opentornado.com)

<http://www.opentornado.com> ~Indonesian High Quality Linux WebStore

### **Referensi**

Script yang digunakan di artikel ini dikembangkan dari artikel di <http://www.thewatchmakerproject.com/journal/276/> dengan berbagai macam perubahan dan perbaikan. Diantaranya adalah perubahan sistem penyimpanan session yang disederhanakan. Selain itu, fitur update shopping cart juga semakin disempurnakan.

## **66. PHP/MySQL Tutorial (Add, View, Edit & Delete from to MySQL database) Live Example (News System )**

### **Introduction**

- While I was learning PHP I searched the net and found a lot of PHP/MySQL tutorials, I went through each, but most of them didn't work as I need to. After weeks I managed to do all about PHP/MySQL. Now I am writing this Tutorial to help you learn how to use PHP and the MySQL database.

- Before you read this tutorial you should have at least a basic knowledge of how to use PHP. If you do not yet know PHP, I suggest that you read this PHP Tutorial

<http://www.freewebmasterhelp.com/tutorials/php/>

During this tutorial we will use live example (News system) and cover these parts:

- Create table called "news" to store the content. (News\_sql.sql)

- Setting up the database name and connection. (config.php)

- Inserting news into the database. (add\_news.php).

- Viewing home news (index.php).

- Viewing the full news (read\_more.php)

- Edit news (edit\_news.php)

- Delete news (delete\_news.php)

So, after you go through this tutorial, you will be able to (Add, View, Edit and Delete from to MySQL database).

In this news system we are going to use the following fields:

1- News title (title).

2- Date & Time (dtime).

3- Home Text (text1).

4- Body Text (text2)

:: download the script files from [here](#)

:: Live example available [here](#)

## \* Part 1 :: Creating database and table (News\_sql.sql)

Code: "insert this code into your database using phpMyAdmin"

```
CREATE TABLE `news` (
    `newsid` int(11) NOT NULL auto_increment,
    `dtime` datetime default NULL,
    `title` varchar(255) default NULL,
    `text1` text,
    `text2` text,
    PRIMARY KEY (`newsid`)
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

The code will create a table that looks like this:

newsid	dtime	title	text1	text2

If we grab a moment and take a look at the SQL-code, and pick it apart we will see that we actually are asking MySQL to do the following:

Please **CREATE** a **TABLE** named **news** with the following columns and properties:

- 1- ( column-name **newsid** with datatype **INT** integer with max 11 digits long, it must be **NOT NULL** and it should be **AUTO\_INCREMENT**,
- 2- column-name **dtime** with datatype **datetime**, it is **NULL**,
- 3- column-name **title** with datatype **VARCHAR** with max (**255**) characters, it is **NULL**,
- 4- column-name **text1** with datatype **text**

5- column-name **text2** with datatype **text**

The **PRIMARY KEY** is (**newsid**),

) **TYPE= MyISAM** MyISAM is the default storage engine as of MySQL 3.23. Each MyISAM table is stored on disk in three files. The files have names that begin with the table name **AUTO\_INCREMENT=1** finally, make the **newsid** auto inserted. End sql-statement ;

Now we're done with the creation of the database and the table, in the next part we will setup the connection and database variables (config.php).

## \* Part 2 :: Connecting to MySQL database (config.php)

Code:

```
<?php  
  
$dbhost="localhost";  
  
$dbusername="user";  
  
$dbpassword="pass";  
  
$dbname="news_system";  
  
  
//Before you can perform any operation on a database you must connect to the MySQL server. The syntax for performing this operation is simple:  
  
$connect = mysql_connect($dbhost, $dbusername, $dbpassword);  
  
mysql_select_db($dbname,$connect) or die ("Could not select database");  
  
?>
```

What I've done here is just to set all the variables, it looks better and it's much easier to see what you are doing once the code starts to get big.

`$dbhost` is a variable and holds your localhost or server name.

`$dbusername`, is a variable and holds your database username

`$dbpassword`, is a variable and holds your username password

`$dbname` is a variable and holds your database name

`$connect` = is a variable and will store the connection for future use.

`mysql_connect` creates a connection to MySQL, it takes three arguments; "host, user and password" it returns false if it fails to connect.

`mysql_select_db` means; please select and open the database. The "`or die()`"-statements is used to display error message if it doesn't find the database.

By this we are end of the connection and selection of the database.

Next we will learn how to add news to the database.

### \* Part 3 :: Add News to MySQL database (add\_news.php)

Code:

```
<?php  
include("config.php");  
  
if(isset($_POST['submit']))  
{//begin of if($submit).  
 // Set global variables to easier names  
 // and prevent sql injection and apostrophe to break the db.  
 $title = mysql_escape_string($_POST['title']);  
 $text1 = mysql_escape_string($_POST['text1']);
```

```

$text2 = mysql_escape_string($_POST['text2']);

//check if (title) field is empty then print error message.

if(!$title){ //this means If the title is really empty.

    echo "Error: News title is a required field. Please fill it.';

    exit(); //exit the script and don't do anything else.

}// end of if

//run the query which adds the data gathered from the form into the database

$result = mysql_query("INSERT INTO news (title, dtime, text1, text2)

VALUES ('$title',NOW(),'$text1','$text2')",$connect);

//print success message.

echo "<b>Thank you! News added Successfully!<br>You'll be redirected to Home Page after (4)
Seconds";

echo "<meta http-equiv=Refresh content=4;url=index.php>";

}//end of if($submit).

// If the form has not been submitted, display it!

else

{//begin of else

?>

<br>

<h3>::Add News</h3>

<form method="post" action="<?php echo $PHP_SELF ?>">

Title: <input name="title" size="40" maxlength="255">

<br>

```

```

Text1:<textarea name="text1" rows="7" cols="30"></textarea>

<br>

Text2:<textarea name="text2" rows="7" cols="30"></textarea>

<br>

<input type="submit" name="submit" value="Add News">

</form>

<?

}//end of else

?>

```

Now lets describe what was that.

- 1- `include("config.php");` is used to load the connection to use it. Remember that we put the connection and variables of database in this file as we explained before.
- 2- `if(isset($_POST['submit']))` : it means if you Clicked the (Add News) button of the form.

Why `$submit`? Because it's the name of the from, look at the form's code here:

`<input type="submit" name="submit" value="Add News">` you see the name of the form is (submit) and of course you can change to whatever you want.

`3- $title = mysql_escape_string($_POST['title']);` this will hold the value of title field from the form

and `mysql_escape_string` will prevent sql injection and apostrophe to break the db.

`$text1 = mysql_escape_string($_POST['text1']);` this will hold the value of the Home text from the form

and `mysql_escape_string` will prevent sql injection

`$text2 = mysql_escape_string($_POST['text2']);` this will hold the value of the Body text from the form

and `mysql_escape_string` will prevent sql injection

`$_POST` is a super global variable of PHP.

4- `if(!$title){` :this means, if the title field is empty then print error message.

5- `exit();` exit the script and don't do anything else or don't go to the next code lines.

6- `$result = mysql_query("INSERT INTO news (title, dtime, text1, text2)`

```
VALUES ('$title',NOW(),'$text1','$text2')",$connect);
```

This may look a little confusing at first, so I'll explain what it all means.

First, `$result` = is there because we are assigning to variable that will run the query.

"`mysql_query()`" takes two arguments, the query and the connection, exactly like this: "`$result = mysql_query($query,$connection);`".

The query is `INSERT INTO news VALUES` : and responsible for inserting the data into a given table's fields.

- `VALUES ('$title',NOW(),'$text1','$text2`

a- `'$title'` : returns the news title you entered in the form.

b- `NOW()`, this function will insert the date and time in the database using the following format: (`2004-10-27 14:05:28`), remember that we used (datatype `datetime`) when we created the table. So if we used `NOW()` the datatype must be datetime.

c- `'$text1'` : returns the news home text you entered in the form.

d- `'$text2'` : returns the news body text you entered in the form.

and `$connect` was defined before in the config.php file, you see we called it here.

\* Congratulations. Every thing is okay! Now print the success message:

```
echo "<b>Thank you! News added Successfully!<br>You'll be redirected to Home Page after (4) Seconds";
```

```
echo "<meta http-equiv=Refresh content=4;url=index.php>";
```

It will print a success message and redirect you to the index.php page after 4 seconds you can change it to whatever you want. It is very simple as you see.

7- `else{`

what is else? As we mentioned before "`if($submit){}`" if the form was submitted, but `else` do the opposite if the form wasn't submitted then show it.

The form:

```
<form method="post" action="<?php echo $PHP_SELF ?>"> and the rest come after this like the inputs and the textareas ... etc. Setting up the HTML form is very easy.
```

Now then, I'm sure you are wondering what the PHP is in there for - that variable, `$PHP_SELF` uses the current URL as the action for the form - so you can put the form handling script in the same document

as the HMTL - which is especially useful when your script isn't very long, as in this case. was that very easy? Now lets go to the next step.

#### \* Part 4 :: A- Viewing Home News (index.php)

Code:

```
<?php  
  
// load the configuration file.  
  
include("config.php");  
  
//load all news from the database and then OREDER them by newsid  
  
//you will notice that newlly added news will appeare first.  
  
//also you can OREDER by (dtime) instaed of (news id)  
  
$result = mysql_query("SELECT * FROM news ORDER BY newsid DESC",$connect);  
  
//lets make a loop and get all news from the database  
  
while($myrow = mysql_fetch_assoc($result))  
  
{//begin of loop  
  
//now print the results:  
  
echo "<b>Title: ";  
  
echo $myrow['title'];  
  
echo "</b><br>On: <i>";  
  
echo $myrow['dtime'];  
  
echo "</i><hr align=left width=160>";
```

```

echo $myrow['text1'];

// Now print the options to (Read,Edit & Delete the news)

echo "<br><a href=\"read_more.php?newsid=$myrow[newsid]\">Read More...</a>
|| <a href=\"edit_news.php?newsid=$myrow[newsid]\">Edit</a>
|| <a href=\"delete_news.php?newsid=$myrow[newsid]\">Delete</a><br><hr>";

}//end of loop

?>

```

Now lets describe what was that?

1- \$result = mysql\_query("SELECT \* FROM news ORDER BY newsid DESC",\$connect);

The script above is using the mysql\_query function to retrieve records or rows from our news table and store them on \$results, then order them by newsid. DESC will print the last inserted record first.

2- while(\$myrow = mysql\_fetch\_assoc(\$result))

-While: do a loop until you get the last record.

- \$myrow: a variable, will hold the records data.

- **mysql\_fetch\_assoc** : Returns an associative array that corresponds to the fetched row and it accesses the data retrieved by the previous mysql\_query stored as \$result. You can use (**mysql\_fetch\_array** instead of **mysql\_fetch\_assoc**).

3- echo "<b>Title: ";

echo \$myrow['title'];

echo "</b><br>On: <i>";

echo \$myrow['dtime'];

\$myrow['title']; is the value of the row which comes from the table, and so for the rest.

print the results and print the options like (Read More...)(Edit)(Delete).

```
- <a href=\"read_more.php?newsid=$myrow[newsid]\">Read More...</a>
```

As you click "Read More..", it will direct you to ([read\\_more.php](#))page where you will see the complete news that you are reading not a random one.

```
- <a href=\"edit_news.php?newsid=$myrow[newsid]\">Edit</a>
```

As you click "Edit", it will direct you to ([edit\\_news.php](#)) page where you edit the news you selected.

```
- <a href=\"delete_news.php?newsid=$myrow[newsid]\">Delete</a>
```

As you click "Delete", it will direct you to ([delete\\_news.php](#)) page where it will delete the news you selected.

## Why we used: \$myrow[newsid]?

Do you remember when we created the news table we created a column-name **newsid** with datatype **INT** integer with max 11 digits long, **AUTO\_INCREMENT=1**, make the **newsid** auto inserted.

Well, each time you add news to the database the **newsid** field will be increased one number, so, for the first time you added news, the **newsid** will become (1) and if you add another it will increase to (2) and then 3 , 4 , 5, 6 and so on.

We use **newsid** to manage the news, so we will depend on the value of the **newsid** because it's an integer not a character and cannot be duplicated because it's automatic.

### \* Part 4 :: B- Viewing Full News (read\_more.php)

Code:

```
<?php
```

```

include("config.php");

$newsid = $_GET['newsid'];

$result = mysql_query("SELECT * FROM news WHERE newsid='$newsid' ",$connect);

while($myrow = mysql_fetch_assoc($result))

{

    echo "<b>";

    echo $myrow['title'];

    echo "</b><br>On: <i>";

    echo $myrow['dtime'];

    echo "</i><hr>";

    echo $myrow['text1'];

    echo " ";

    echo $myrow['text2'];

    echo "<br><br><a href=\"javascript:self.history.back();\"><-- Go Back</a>";

}

?>

```

#### Description:

1- \$result = mysql\_query("SELECT \* FROM news WHERE newsid='\$newsid' ",\$connect);

This query will select only the news you meant to read, not a random one, suppose you're reading a news about security and you clicked on Read More.. then it must direct you to the rest of the security article you are reading not any other one. Here **newsid** plays its rule. As you see **SELECT \* FROM news**

**WHERE newsid='\$newsid'** we selected from news table only the article that matched the **newsid** value. Suppose the security article has the number (3) then the value of the **newsid** must be (3).

You will understand this after you add news to the database and view it. Before you click on read more... watch the stats bar. You will see something like ([http://localhost/news/read\\_more.php?newsid=3](http://localhost/news/read_more.php?newsid=3)) and after you click, give a look at the address bar.

```
2-    while($myrow = mysql_fetch_assoc($result)) {  
  
        echo "<b>";  
  
        echo $myrow['title'];  
  
        echo "</b><br>On: <i>";  
  
        echo $myrow['dtime'];  
  
        echo "</i><hr>";  
  
        echo $myrow['text1'];  
  
        echo " ";  
  
        echo $myrow['text2'];  
  
        echo "<br><br><a href=\"javascript:self.history.back();\"><-- Go Back</a>";  
  
    }  
  
}
```

This will print only the selected news. Note there is: **javascript:self.history.back()**; this is java. When you click on go back it will direct you to the previous page.

Was that fun?!!! I think so. Hey ;) lets move next.

## \* Part 5 :: Editing News (edit\_news.php)

Code:

```
<?php

include("config.php");

if(isset($_POST['submit']))

{

    // Set global variables to easier names

    // and prevent sql injection and apostrophe to break the db.

    $title = mysql_escape_string($_POST['title']);

    $text1 = mysql_escape_string($_POST['text1']);

    $text2 = mysql_escape_string($_POST['text2']);



    $result = mysql_query("UPDATE news SET title='$title', text1='$text1', text2='$text2' WHERE newsid='$newsid' ",$connect);





    echo "<b>Thank you! News UPDATED Successfully!<br>You'll be redirected to Home Page after (4) Seconds";

    echo "<meta http-equiv=Refresh content=4;url=index.php>";

}

elseif(isset($_GET['newsid']))
```

```

{

$result = mysql_query("SELECT * FROM news WHERE newsid='$_GET[newsid]' ", $connect);

while($myrow = mysql_fetch_assoc($result))

{

$title = $myrow["title"];

$text1 = $myrow["text1"];

$text2 = $myrow["text2"];



?>

<br>

<h3>::Edit News</h3>




<form method="post" action="<?php echo $PHP_SELF ?>">

<input type="hidden" name="newsid" value="<? echo $myrow['newsid']?>">




Title: <input name="title" size="40" maxlength="255" value="<? echo $title; ?>">

<br>

Text1: <textarea name="text1" rows="7" cols="30"><? echo $text1; ?></textarea>

<br>

Text2: <textarea name="text2" rows="7" cols="30"><? echo $text2; ?></textarea>

<br>

<input type="submit" name="submit" value="Update News">

```

```
</form>

<?

    }//end of while loop
```

```
}//end else
```

```
?>
```

What was that?

This script exactly the same as (add\_news.php) but here we update a previously added news.

1- `include("config.php");` : loads the connection.

2- `if($submit)` : if you clicked on (**Update News**) button.

`$title = mysql_escape_string($_POST['title']);`

`$text1 = mysql_escape_string($_POST['text1']);`

`$text2 = mysql_escape_string($_POST['text2']);`

Define easy variables to hold the value of the form fields and prevent sql injections.

4- `$result = mysql_query("UPDATE news SET title='$title', text1='$text1', text2='$text2' WHERE newsid='$newsid' ",$connect);`

This query updates a specified news depending on the `newsid` value.

5- `elseif($newsid):` if the form wasn't submitted then view the news before updating. Here we used the value of `newsid` to drag the data from the database in put it in the form fields.

6- `<input type="hidden" name="newsid" value="<? echo $myrow['newsid']?>">`

Is used to print the news in the form fields in-order to modify.

```
- <input type="submit" name="submit" value="Update News"></form>
```

The end of the form.

## \* Part 6 :: Deleting News (delete\_news.php)

Code:

```
<?php  
  
include("config.php");  
  
$newsid = $_GET['newsid'];  
  
$result = mysql_query("DELETE FROM news WHERE newsid='$newsid' ",$connect);  
  
  
echo "<b>News Deleted!<br>You'll be redirected to Home Page after (4) Seconds";  
  
//header("location: index.php");  
  
echo "<meta http-equiv=Refresh content=4;url=index.php>";  
  
?>
```

The final part which needs to be created is a page to delete news. As with the edit news page we depend on the **newsid** and when you click on delete a URL like **delete\_news.php?newsid=3** will be shown to you. The code to do this is the same as to edit news, except with a slightly different MySQL query. Instead of the UPDATE query you should use: **DELETE FROM news WHERE newsid='\\$newsid'** .

There we go, a nice simple method to create a news system!

Thank you for your time reading my tutorial. I hope you understand it and got the maximum benefit.

**Author: Mohammed Ahmed**

**Country: Palestine, Gaza**

**Email/MSN: m (at) maaking.com**

**Web: <http://www.maaking.com>**

## 67. Creating A Menu System

In this tutorial I will show you how to create a simple menu system with 2 levels. You can easily integrate it into your site to get a nice and easy editable site navigation system.

### Creating a menu system

In this tutorial I will show you how to create a simple menu system with 2 levels. You can easily integrate it into your site to get a nice and easy editable site navigation system.

#### Step 1.

To make a menu system easy changeable and easy readable we will separate the code into more parts. A basic realization contains 3 files which are the following:

- menuStruct.php: Contains only the navigation structure. It will be used by the handler function.
- menuHandler.php: This file contains the PHP code which generates the menus with corresponding submenus as well.
- index.php: The main page which integrates the navigation system.

#### Step 2.

First let's design the menu system structure. To handle the menu element easy we will use a multi dimensional array. In the array the key will be the name of the menu and the value is the link for it. So we can define the main menu system as follows:

```
<?php
    // Main menu items
    $mainMenu['Home']    = 'link-1';
    $mainMenu['About us'] = 'link-2';
    $mainMenu['Projects'] = 'link-3';
    $mainMenu['Contact']  = 'link-4';
    $mainMenu['Others']   = 'link-5';
?>
```

Now we have all the main menu items defined. However usually we need some submenus as well. To store it in the array we will create a new array which stores the main menu - submenu relations. As you will see here we use a little bit more complex array to store information. You have to take care that the keys - key strings - in the main array and in the sub array must be identical. So the submenu array looks like this:

```
<?php
    // Sub menu items
    $subMenu['Projects'][SubMenu-1] = 'sub-link-1';
```

```

$subMenu['Projects']['SubMenu-2'] = 'sub-link-2';
$subMenu['Projects']['SubMenu-3'] = 'sub-link-3';
$subMenu['Projects']['SubMenu-4'] = 'sub-link-4';

$subMenu['Others']['SubMenu-1'] = 'sub-link-11';
$subMenu['Others']['SubMenu-2'] = 'sub-link-12';
$subMenu['Others']['SubMenu-3'] = 'sub-link-13';
?>

```

At the end store this information in a file let's named it to menuStruct.php.

### Step 3.

After you stored the menu structure in a file we can focus on the application logic. We will implement it in the second file named menuHandler.php. In this file we will implement only one function called **createMenu()**. This function will have only one parameter which tells what is the actual page link. From this information the function will decide whether or not to display relevant submenus. So the final result is this:

```

<?php
function createMenu($actLink){

}
?>

```

Now let's start the most important part. First the function needs to include the menuStruct.php file to access the menu arrays. As second we open a table tag where we will display the menu elements as rows (tr tags).

As next step we check where the input parameter - which is the actual link - can be found in the arrays. It is important to know to decide which submenu the code should display. So the code is the following:

```

<?php
// Get actual Main menu
$actMenu="";
foreach ($mainMenu as $menu => $link) {
    if ($link == $actLink) $actMenu = $menu;
    if (isset($subMenu[$menu])){
        foreach ($subMenu[$menu] as $menuSub => $linkSub) {

```

```

        if ($linkSub == $actLink) $actMenu = $menu;
    }
}
}
?>

```

After we got the actual menu we can begin with displaying the menu items row by row. We start it with the main menu items. In all steps we check whether the actually selected link in the array and the menu is the same or not. It is because if they are the same then we should display the relevant submenus before displaying the next main menu item. To make it more pretty we use other CSS class for the submenu rows as the main menu rows. The code looks like this:

```

<?php
foreach ($mainMenu as $menu => $link) {
    echo '<tr><td><a href="'. $link.'">'.$menu.'</a></td></tr>';
    if ( ($actMenu == $menu) && (isset($subMenu[$menu])) ) {
        foreach ($subMenu[$menu] as $menuSub => $linkSub) {
            echo '<tr><td><a href="'. $linkSub.'" class="submenu">'.$menuSub.'</a></td></tr>';
        }
    }
}
?>

```

#### Step 4.

The last step is a demonstration how to use it in the real life. To do this we will create a new file. Let's called it to index.php. This is quite straightforward. We include the menuHandler.php and create our HTML page as we want. Where you want to display the navigation block you only need to call our function with the actual link as

```
<div id="menu"><?php createMenu('link-3'); ?></div>
```

## 68. Membuat Format Rupiah di PHP

Jika temen2 ada yang mencari cara untuk membuat format mata uang khususnya rupiah agar mudah dibaca, ni aq bagi codingnya :

```
<? echo "Rp".number_format($baris[anggaran], 2 , ',' , '.'); ?>
```

Untuk memahaminya, fungsi number\_format dapat dituliskan sebagai berikut :  
number\_format(\$angka,jlm\_desimal,'pemisah desimal','pemisah angka'). Apabila nilai  
\$baris[anggaran] = 1000000 maka nanati akan tampil Rp 1.000.000,00

Atau mungkin temen2 bisa membuat fungsi sendiri seperti dibawah ini :

```
<?php
function rupiah($uang)
{
    $rupiah = "";
    $panjang = Strlen($uang);

    while ($panjang > 3)
    {
        $rupiah = "." . substr($uang, -3) . $rupiah;
        $lebar = strlen($uang) - 3;
        $uang = substr($uang, 0, $lebar);
        $panjang= strlen($uang);
    }

    $rupiah = "Rp".$uang.$rupiah.",-";
    return $rupiah;
}
?>
```

Source ini aq jalanin menggunakan phptriad 1.5.2

Semoga bermanfaat

**69.** P

**70.**      **s**