



EBOOK

# BELAJAR PHP & MySQLi

DARI DASAR SAMPAI MEMBUAT  
SISTEM INFORMASI LAUNDRY

- + LOGIN MD5
- + NOTIFIKASI LOGIN & LOGOUT
- + PENGATURAN HARGA LAUNDRY / Kg
- + PROSES TRANSAKSI
- + KONFIRMASI STATUS TRANSAKSI
- + CETAK INVOICE LAUNDRY
- + FILTER LAPORAN PER TANGGAL
- + PRINT LAPORAN
- + CETAK LAPORAN PDF
- + FULL SOURCE CODE



Diki Alfarabi Hadi, S.T.

WWW.MALASNGODING.COM

“ Ilmu tanpa amal adalah kegilaan, dan amal tanpa ilmu  
adalah kesia-siaan ”.

- Imam Ghazali

# TENTANG PENULIS



Diki Alfarabi Hadi, seorang praktisi IT yang mengkhususkan diri di bidang *web development* dan *mobile app*, memulai petualangan dalam dunia pemrograman sejak tahun 2014 atau semenjak masih kuliah. Lebih senang dengan dunia pemrograman dibanding bidang IT yang lain. Senang berbagi pengetahuan dan mempelajari sesuatu yang baru khususnya bidang web development.

Memiliki pengalaman sebagai programmer di CV. Carikode Indonesia, sebagai *Back-end web developer* di PT. Bin Hasan Indonesia dalam pengembangan sistem crowdfunding, dan beberapa perusahaan lain. Pernah menjuarai beberapa perlombaan *web design* tingkat nasional bersama tim. Salah satunya adalah *web design competition* yang diadakan oleh PT. Pelindo III di Surabaya.

Saat ini penulis aktif berbagi tutorial pemrograman di [www.malasngoding.com](http://www.malasngoding.com). Untuk mengetahui informasi lebih lanjut tentang penulis, silahkan mengunjungi <https://www.malasngoding.com>. Atau mengirimkan email ke [malasngoding@gmail.com](mailto:malasngoding@gmail.com).

# PENGANTAR

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Panyayang, Penulis panjatkan puja dan puji syukur atas kehadirat-Nya, yang telah melimpahkan rahmat, hidayah, dan inayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan buku “ *Belajar PHP & MySQLi Dari Dasar Sampai Membuat Sistem Informasi Laundry* ” ini.

Penulis menyadari sepenuhnya masih ada kekurangan baik dari segi susunan kalimat maupun tata bahasa. Oleh karena itu, dengan tangan terbuka penulis menerima segala saran dan kritik dari pembaca agar menjadi lebih baik untuk ke depannya.

Akhir kata penulis berharap buku ini dapat memberikan manfaat maupun inspirasi terhadap pembaca.

## **DAFTAR ISI**

<b>BELAJAR PHP TINGKAT DASAR .....</b>	<b>1</b>
Belajar PHP Part 1 : Pengenalan PHP .....	2
Pengenalan PHP .....	2
Fungsi atau Kegunaan PHP .....	3
Penulisan Syntax PHP .....	3
Cara menjalankan PHP pada Localhost .....	4
Belajar PHP Part 2 : Instalasi Web Server Di Windows.....	5
Pengertian Web Server .....	5
Pengertian Localhost .....	5
Pengertian XAMPP .....	6
Cara Instalasi Web Server di Windows .....	6
Cara Instalasi XAMPP di Windows .....	6
Belajar PHP Part 3 : Belajar PHP Dasar Untuk Pemula .....	9
Belajar PHP Part 4 : Mengenal Variabel Di PHP .....	12
Pengertian Variabel Pada PHP .....	12
Cara Penulisan Variabel di PHP .....	12
Mengupdate isi variabel .....	14
Belajar PHP Part 5 : Tipe Data Pada PHP .....	16
Tipe Data String pada PHP.....	16
Tipe Data Integer pada PHP .....	17
Tipe Data Float pada PHP .....	17
Tipe Data Boolean pada PHP .....	17
Tipe Data Array pada PHP .....	18
Belajar PHP Part 6 : Manipulasi String Pada PHP .....	19
Menghitung jumlah karakter dengan strlen().....	19
Menghitung jumlah kata pada String dengan str_word_count() .....	20
Membalik String Dengan strrev() .....	20

Memfilter dan mengganti string dengan str_replace() .....	21
Belajar PHP Part 7 : Kondisi IF ELSE Pada PHP .....	23
Mengenal Kondisi If-Else pada PHP .....	23
Mengenal Kondisi If-Elseif-Else.....	25
Belajar PHP Part 8 : Mengenal Function Pada PHP .....	27
Belajar PHP Part 9 : Penggunaan Switch Case Pada PHP.....	30
Belajar PHP Part 10 : Perulangan While Pada PHP .....	33
Belajar PHP Part 11 : Perulangan For Pada PHP .....	35
Belajar PHP Part 12 : Mengenal Array Pada PHP .....	37
Membuat dan memanggil data pada array secara biasa .....	37
Belajar PHP Part 13 : Menampilkan Tanggal Pada PHP .....	42
Cara menampilkan tanggal di PHP .....	42
Menampilkan waktu di PHP .....	44
Belajar PHP Part 14 : Include Dan Require Pada PHP.....	46
Cara penulisan dan penggunaan Include dan Require .....	46
Contoh penggunaan fungsi include dan require PHP.....	47
Perbedaan Include, Require dan Include_once, require_once .....	48
Belajar PHP Part 15 : Penanganan Form Pada PHP.....	50
Membuat penanganan form dengan method GET.....	50
Membuat penanganan form dengan method POST .....	53
Membuat penanganan form dengan \$_REQUEST .....	55
Belajar PHP Part 16 : Operator Aritmatika di PHP .....	56
Cara membuat operator aritmatika di PHP .....	57
Belajar PHP Part 17 : Pengertian dan Pengenalan Foreach.....	59
Cara Penggunaan Foreach di PHP .....	59
Belajar PHP Part 18 : Explode Dan Implode Pada PHP .....	62
Memecahkan string dengan fungsi explode().....	62
Menggabungkan String dengan implode() .....	64

<b>BELAJAR PHP TINGKAT MENENGAH .....</b>	<b>67</b>
Membuat Kalkulator Sederhana Dengan PHP .....	68
Membuat Perhitungan Laba Dan Rugi Dengan PHP .....	75
Membuat Format Tanggal Indonesia Dengan PHP .....	79
Mengubah Format Tanggal Dengan PHP .....	79
Membuat Format Hari Pada PHP .....	83
Penjelasan Membuat Format Hari Indonesia Dengan PHP.....	85
Membuat Format Rupiah Di PHP .....	87
Membuat Fungsi Terbilang Dengan PHP.....	90
Penjelasan .....	92
Penjelasan Function PHP.....	93
Membuat Halaman Web Dinamis Dengan PHP .....	94
Kegunaan Halaman Web Dinamis ?.....	94
Membuat Halaman Web Dinamis Dengan PHP .....	94
Penjelasan .....	100
Membuat Form Validasi dengan PHP .....	103
Membuat Captcha dengan PHP .....	108
Kegunaan captcha.....	108
Cara Membuat Captcha dengan PHP .....	109
Membuat dan Menghapus Folder Dengan PHP .....	115
Kegunaan mkdir() untuk membuat folder atau directory baru .....	115
Kegunaan rmdir() untuk menghapus folder atau directory .....	117
Mengenal MD5 Pada PHP.....	119
Pengertian MD5 .....	119
Kegunaan MD5 .....	119
Cara Membuat MD5 Pada PHP .....	120
Membuat Koneksi Database PHP dan MySQL .....	122
Mempersiapkan Database .....	122

Menghubungkan PHP Dengan Database MySQL .....	123
Membuat Login dengan php dan mysql menggunakan enkripsi md5.....	126
Persiapan Database dan Table .....	126
Membuat Form Login Sederhana .....	128
Membuat validasi login dengan PHP dan mysql .....	133
Login Dengan Enkripsi Password menjadi md5.....	137
Membuat CRUD Dengan PHP dan MySQL.....	139
Pengertian CRUD (Create, Read, Update dan Delete) .....	139
Persiapan & Membuat Koneksi Database .....	140
Menampilkan Data Dari Database .....	142
Input Data Ke Database.....	148
Edit & Update Data Dari Database .....	152
Menghapus Data Dari Database .....	160
Membuat Form Pencarian dengan PHP dan MySQL.....	163
Penjelasan .....	168
Membuat Upload File Dengan PHP Dan MySQL.....	170
Persiapan Membuat Upload File.....	170
Membuat Upload File .....	171
Membuat Pagination atau Penomoran Halaman Dengan PHP dan MySQL .....	179
Penjelasan .....	180
Koneksi Database PHP dan MySQLi .....	184
Kenapa Menggunakan MySQLi ? .....	184
Menghubungkan PHP dengan database menggunakan MySQLi .....	184
Membuat CRUD dengan PHP dan MySQLi .....	189
Persiapan Database .....	189
Koneksi Database PHP dan MySQLi.....	191
Menampilkan Data Dari Database .....	192
Menginput Data Ke Database .....	196

Edit & Update Data Pada Database .....	201
Hapus Data Dari Database .....	208
Membuat Login Dengan PHP dan MySQLi.....	210
Persiapan .....	210
Membuat Koneksi Database.....	213
Membuat Login PHP dan MySQLi .....	214
Membuat Halaman Admin .....	218
Membuat Login Dengan PHP dan MySQLi Menggunakan MD5 .....	225
Persiapan .....	225
Membuat Login Dengan PHP dan MySQLi Menggunakan MD5 .....	227
<b>BELAJAR PHP TINGKAT LANJUTAN – MEMBUAT SISTEM INFORMASI LAUNDRY ..</b>	<b>233</b>
Persiapan dan Penjelasan .....	234
Membuat Project Aplikasi PHP Baru.....	235
Persiapan Database .....	235
Instalasi Bootstrap .....	238
Instalasi Bootstrap .....	238
Instalasi Jquery .....	240
Koneksi Database.....	240
Membuat Login Admin.....	242
Membuat Tabel Admin .....	242
Membuat Halaman Login.....	244
Membuat Aksi Login Admin .....	247
Membuat Logout Admin.....	252
Membuat Halaman Admin (Halaman Dinamis) .....	255
Membuat Ganti Password Admin.....	261
Membuat CRUD Pelanggan Dengan PHP dan MySQLi .....	266
Membuat Tabel Pelanggan .....	266
Menampilkan Data Pelanggan Dari Database.....	269

Menginput Data Pelanggan Ke Database.....	271
Mengedit Data Pelanggan.....	276
Menghapus Data Pelanggan.....	281
Membuat Pengaturan Harga Laundry Per Kilo .....	283
Membuat Table Harga.....	283
Membuat Pengaturan Harga Laundry .....	286
Membuat Transaksi Laundry .....	290
Membuat Table Transaksi.....	290
Membuat Table Pakaian .....	293
Menampilkan Data Transaksi Laundry .....	294
Menginput Data Transaksi Laundry Baru.....	298
Membuat Edit Data Transaksi Laundry .....	309
Membuat Invoice Transaksi Laundry .....	318
Membuat Cetak Invoice Transaksi Laundry.....	322
Membuat Batalkan Transaksi Laundry.....	327
Membuat Laporan Laundry.....	330
Membuat Filter Laporan.....	330
Membuat Print Laporan .....	337
Membuat Cetak Laporan Ke PDF dengan DOMPDF .....	342
Rekapitulasi Data Pada Dashboard Admin .....	347

# **BELAJAR PHP TINGKAT DASAR**

# Belajar PHP Part 1 : Pengenalan PHP

Selamat datang di tutorial PHP dasar Belajar PHP Part 1 : Pengenalan PHP. anda yang sudah mulai memasuki tutorial dasar php ini diasumsikan untuk mempelajari tentang HTML dan CSS dasar terlebih dahulu sebelum melompat ke tutorial PHP. untuk tutorial HTML dan CSS dasar nya sendiri bisa anda ikuti di tutorial-tutorial www.malasngoding.com. Akan sangat lebih baik jika bahasa pemrograman di pelajari secara bertahap agar mengerti dari tiap dasar dan fungsi nya masing-masing. hal ini dapat mengakibatkan anda menemukan kesulitan dalam mempelajari pemrograman php jika HTML dan CSS dasar belum anda kuasai. sehingga dapat berujung kepada malas ngoding.

Pada sub judul Belajar PHP Part 1 : Pengenalan PHP ini akan di jelaskan tentang pengenalan-pengenalan dari bahasa pemrograman PHP. seperti bagaimana bentuk penulisan syntax PHP, sifat dari PHP, struktur nya dan bagaimana cara menjalankan file PHP. berikut tentang pengenalan dasar bahasa pemrograman PHP.

## Pengenalan PHP

PHP adalah singkatan dari *Hypertext Preprocessor* dan merupakan bahasa pemrograman yang didesain khusus untuk *web development* atau pengembangan web. PHP memiliki sifat *Server-Side*. karena PHP dijalankan atau di eksekusi dari sisi server. maksud dijalankan dari sisi server adalah PHP dijalankan pada komputer server dan bukan pada komputer client. PHP dijalankan melalui aplikasi web browser sama halnya seperti HTML. Hampir semua situs-situs besar dan populer dikembangkan menggunakan PHP. seperti misalnya wordpress, joomla, facebook, twitter, wikipedia dan situs besar lainnya.

PHP mulai di kembangkan pada tahun 1995 oleh Rasmus Lerdorf. untuk situs resmi dari PHP sendiri dapat di akses pada url <http://www.php.net>. Setelah anda mempelajari tentang HTML dan CSS dasar di malasngoding.com tentu anda sudah dapat mengambil kesimpulan bahwa HTML di gunakan untuk membuat pondasi pada sebuah website, dan CSS di gunakan untuk men-design halaman website. maka pada sub judul tutorial belajar PHP dasar ini anda akan mengenal apa itu PHP dan bagaimana kegunaannya dalam membangun sebuah website yang memiliki halaman statis dan sebagainya.

## Fungsi atau Kegunaan PHP

Jika di ibaratkan pada sebuah pondasi bangunan rumah untuk gambaran fungsi atau kegunaan PHP, kita bisa menyebut HTML sebagai tiang-tiang pondasi, atap, lantai dan dinding. di sini kita belum membicarakan tentang warna, bentuk pintu, luas ruangan. misanya jika pondasi, dinding dan struktur bangunan rumah sudah di buat tentu bangunan rumah tersebut masih tampak sangat monoton dan tidak sedap di pandang mata. oleh karena itulah kita menggunakan CSS untuk membuat bentuk dan tampilan rumah tersebut menjadi menarik dan enak di pandang. nah, pada contoh kasus bangunan rumah ini, pasti anda bertanya-tanya PHP di gunakan sebagai apa ? pada contoh kasus bangunan rumah ini PHP di gunakan untuk membuat siapa saja yang dapat masuk ke dalam rumah, siapa yang dapat melihat rumah, dan siapa yang dapat menyimpan memasukkan atau membuang barang-barang pada rumah tersebut. jadi intinya PHP berguna untuk membuat aksi pada rumah atau website jika pada contoh nyatanya.

## Penulisan Syntax PHP

Untuk penulisan syntax nya, PHP di tandai dengan membuat tag pembuka (`<?php`) dan di akhiri dengan tag penutup (`?>`). syntax php dapat di sisipkan pada bagian-bagian HTML. kemudian di akhir setiap baris syntax PHP harus di tutup dengan tanda semicolon atau titik koma`(;)`berikut ini adalah contoh penulisan syntax php yang benar.

```
<?php  
echo "Belajar Pemrograman PHP di malasngoding.com";  
?>
```

Dapat anda lihat pada contoh penulisan syntax PHP di atas, bahwa pada awal syntax php harus di buka dengan tag pembuka php yang di tandai dengan “`<?php`”. dan di tutup dengan tag penutup PHP “`?>`”. dan yang perlu di perhatikan lagi adalah di akhir baris syntax PHP harus di akhiri atau di tutup dengan semicolon atau titik koma`(;)`.

Syntax echo pada contoh di atas memiliki fungsi atau kegunaan sebagai penampil data. adapun output yang di hasilkan pada contoh syntax php di atas adalah tulisan “ Belajar Pemrograman PHP di malasngoding.com ”.

## Cara menjalankan PHP pada Localhost

Setelah membaca tutorial di atas tentang cara penulisan syntax PHP. selanjutnya akan di jelaskan juga cara menjalankan syntax PHP pada localhost. apa itu localhost ? bisa di lihat pada tutorial belajar PHP dasar selanjutnya.

## Belajar PHP Part 2 : Instalasi Web Server Di Windows

Pada artikel bagian kedua di Tutorial Belajar PHP dasar ini akan di jelaskan tentang cara instalasi Web Server atau XAMPP untuk membuat localhost. karena PHP harus di jalankan dari sisi server atau dari komputer server, maka anda perlu menginstall web server di PC atau laptop anda. untuk menjadikan laptop atau pc anda sebagai web server yang mampu menjalankan syntax atau file PHP. jadi pada tutorial Belajar PHP Part 2 : Instalasi Web Server Di Windows ini akan di jelaskan tentang cara instalasi web server di laptop atau pc. kebetulan contoh pada tutorial instalasi web server ini menggunakan sistem operasi windows 7. tapi jangan cemas, bagi anda yang menggunakan sistem operasi linux juga dapat melihat tutorial cara instalasi web server di linux di [malasngoding.com](http://malasngoding.com).

### Pengertian Web Server

Sebelum masuk ke tahap tutorial cara menginstal web server akan di jelaskan terlebih dahulu tentang pengertian web server. Web Server adalah layanan yang memiliki fungsi untuk *received* (menerima) *request* atau permintaan dari HTTP/HTTPS dari client melalui *web browser* (Google Chrome, Mozilla Firefox, Opera dan lain-lain) dan mengirimkan kembali hasil *request* kepada *client* berupa halaman website yang umumnya berbentuk file .HTML atau .PHP.

Ada beberapa contoh jenis web server yang paling populer dan paling banyak di gunakan yaitu Apache dan Microsoft Windows Server milik dari Perusahaan Microsoft.

### Pengertian Localhost

Localhost merupakan server local atau *web server* yang bekerja atau berjalan pada laptop atau pc anda. alamat IP dari localhost adalah 127.0.0.1 yang kemudian di terjemahkan menjadi LOCALHOST. jadi localhost terletak pada pc atau laptop anda. anda membutuhkan localhost untuk menjalankan file .php sebagaimana yang telah di jelaskan sebelum nya di tutorial dasar php pengenalan php bahwa PHP hanya dapat berjalan pada sisi server oleh karena itu PHP di sebut sebagai Pemrograman Server Side atau berada pada sisi server.

localhost dijadikan sebagai server sementara pada saat pengembangan aplikasi yang berbasis website sebelum di hosting kan atau di online kan. localhost

hanya dapat di akses dari laptop atau pc anda sendiri dengan cara mengakses langsung pada web browser dengan alamat IP dari localhost yaitu `http://127.0.0.1` atau bisa juga dengan mengakses `http://localhost`. maka halaman akan di alihkan ke localhost tempat anda membuat project aplikasi yang berbasis website. Jadi inti dari localhost adalah server local yang di pasang pada pc atau laptop anda untuk dapat menjalankan file PHP karena PHP memiliki sifat server side atau bekerja pada sisi server.

## Pengertian XAMPP

XAMPP adalah sebuah *software* (perangkat lunak) untuk menginstall atau memasang localhost pada pc atau laptop. aplikasi XAMPP sendiri bersifat gratis atau *free*. sehingga banyak para pengembang web (*Web Developer*) menggunakan XAMPP untuk menginstall localhost dan mysql karena selain gratis, xampp juga sangat powerfull dalam *menagement* data dan cara penginstalannya.

Dengan menginstall XAMPP maka database MySQL dan phpmyadmin juga akan ikut terinstal. karena anda akan membutuhkan MySQL untuk penyimpanan database website anda. dan dapat mengelola database dengan mudah dan instan melalui PhpMyAdmin.

## Cara Instalasi Web Server di Windows

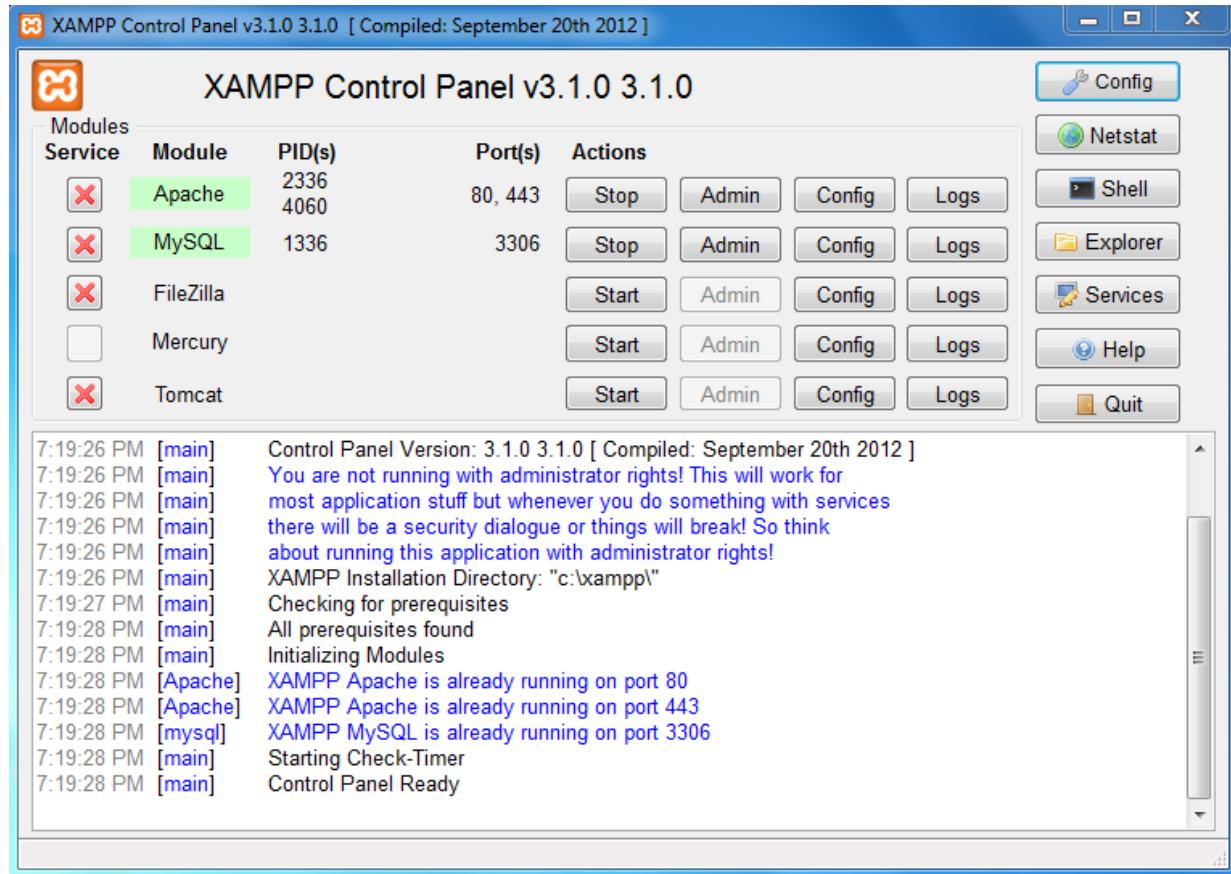
Untuk cara penginstalan web server di windows ini sebagai contohnya akan menggunakan aplikasi XAMPP. sebenarnya tidak hanya XAMPP yang memiliki fungsi untuk menginstall localhost, ada juga WAMPP dan LAMPP untuk linux. tetapi pada contoh cara instalasi *web server* di windows ini kita akan menggunakan XAMPP.

## Cara Instalasi XAMPP di Windows

Untuk tahap awal cara penginstalan XAMPP anda harus men-download terlebih dahulu. untuk men-download XAMPP anda dapat men-downloadnya langsung dari website resminya. <https://www.apachefriends.org/download.html>

Kemudian double klik pada master XAMPP yang sudah di download. Kemudian anda hanya tinggal mengikuti intruksi penginstalan XAMPP nya langsung seperti cara menginstall aplikasi windows lainnya.

Nah, jika anda sudah melakukan penginstalan XAMPP sesuai prosedur maka langkah selanjutnya adalah memeriksa apakah XAMPP sudah berjalan dengan baik ? cara nya buka shortcut XAMPP.



### Instalasi Web Server Di Windows

Seperi pada gambar di atas. klik pada tombol start pada Apache untuk mengaktifkan localhost dan MySQL untuk mengaktifkan database MySQL. kemudian ketikkan <http://localhost> atau bisa juga <http://127.0.0.1> pada web browser yang anda gunakan. maka akan muncul intruksi penggunaan XAMPP sebagai berikut:



### cara install web server di windows

Maka instalasi xampp pun berhasil dan berjalan tanpa masalah. sekian lah Belajar PHP Part 2 : Instalasi Web Server Di Windows. selanjutnya pada Belajar PHP Part 3 anda akan mulai mempelajari cara penulisan syntax php dasar dan cara menjalankannya di localhost.

## Belajar PHP Part 3 : Belajar PHP Dasar Untuk Pemula

Belajar PHP Dasar Untuk Pemula, sesuai judul pada artikel ketiga dari tutorial belajar php ini anda akan belajar tentang dasar-dasar dari PHP, dari mulai cara penulisan syntax PHP, kemudian di simpan dengan ekstensi apa, dan bagaimana cara menjalankannya di localhost atau server lokal laptop atau pc. bagi anda yang belum membaca tutorial sebelumnya tentang pengenalan PHP dan cara instalasi web server alangkah lebih baik jika membacanya terlebih dahulu.

Untuk tahap awal mulai belajar php akan di jelaskan terlebih dahulu tentang cara penulisan syntax php, syntax PHP harus di tuliskan di dalam tag php. syntax php di awali dengan tag pembuka PHP " <?php " dan di akhiri dengan syntax penutup PHP " ?> ". dan yang paling penting di setiap akhir baris dari syntax php harus di akhiri dengan semicolon ";" atau titik koma sebagai tanda bahwa baris syntax PHP tersebut sudah selesai. untuk langkah selanjutnya kita akan mencoba beberapa syntax dasar php untuk mencoba menjalankan PHP di localhost anda. silahkan buka text editor anda dan ketik syntax berikut:

```
<?php  
echo "Halo, selamat belajar di www.malasngoding.com";  
?>
```

Seperti yang di lihat pada syntax di atas. penulisan syntax PHP harus dibuka dengan tag pembuka dan ditutup dengan tag penutup. dan di akhir baris syntax php harus di akhiri dengan tanda semicolon atau titik koma. ada juga orang yang menulis syntax php dengan "<?" dan di akhiri dengan "?>". Belajar PHP Belajar PHP Dasar Untuk Pemula. sebenarnya itu adalah contoh penulisan syntax yang salah karena tidak sesuai dengan aturan dasar dari PHP sendiri. jadi sebaiknya mengikuti cara penulisan tag php secara benar sesuai aturan dasar dari PHP yaitu "<?php" dan tag penutup nya "?>" .

```
<?php —— Tag Pembuka  
echo "Halo, selamat belajar di www.malasngoding.com";  
?>  
|————— Di akhiri dengan semicolon  
Tag Penutup
```

MALAS  
NGODING  
COM

Belajar PHP dasar untuk pemula

Selanjutnya simpan file tersebut dengan menggunakan ekstensi .php(dot php). di asumsikan untuk pembelajaran dasar anda boleh menyimpannya dengan nama belajar.php . simpan ke direktori localhost anda seperti saat penginstalan. umumnya banyak orang yang menyimpan atau menginstallnya di C://xampp/htdocs/ bagi anda yang menggunakan xampp di windows dan di /var/www/ jika anda yang menggunakan linux. oleh sebab itu simpan di folder tersebut file belajar.php yang sudah di buat tadi. kemudian aktifkan apache anda seperti yang sudah di jelaskan pada tutorial belajar PHP dasar part 2. dan buka localhost. kemudian ketikkan alamat localhost anda dan file php yang ingin di eksekusi. misalnya pada contoh ini anda bisa mengetikkan <http://localhost/belajar.php> atau boleh juga dengan menggunakan alamat IP localhost <http://127.0.0.1/belajar.php>. dan hasil nya :



Halo, selamat belajar di [www.malasngoding.com](http://www.malasngoding.com)

MALAS  
NGODING  
COM

Belajar PHP Belajar PHP Dasar Untuk Pemula

Dari contoh di atas dapat di lihat bahwa kita telah berhasil menjalankan file PHP yang kita buat di localhost.

## Belajar PHP Part 4 : Mengenal Variabel Di PHP

Sampai pada tutorial Belajar PHP Part 4 : Mengenal Variabel Di PHP. Setelah pada tutorial part 1 sampai part 3 anda sudah mengetahui bayangan-bayangan tentang apa itu PHP, cara penulisan PHP yang benar dan cara menginstall XAMPP dan menjalankan PHP di localhost. Pada tutorial Belajar PHP Part 4 ini kita akan belajar tentang mengenal apa itu variabel dan bagaimana cara penulisan variabel pada bahasa pemrograman PHP.

### Pengertian Variabel Pada PHP

Sama seperti pada bahasa pemrograman lainnya yang memiliki variabel sebagai tempat atau wadah untuk menyimpan data sementara. variabel bisa dikatakan sebagai penyimpanan informasi atau data sementara di dalam pemrograman. variabel bisa menyimpan data berupa integer, string dan lainnya. nilai atau isi dari pada variabel sendiri bisa di update atau di ubah yang akan kita praktekkan di dalam tutorial belajar php mengenal variabel di php ini. variabel memiliki nama nya masing-masing dengan tujuan untuk memudahkan programmer untuk menandai masing-masing variabel tersebut. dan memudahkan untuk pemanggilan variabel nya tersebut.

### Cara Penulisan Variabel di PHP

Penulisan variabel di php mempunyai ketentuan tersendiri. pada sub judul ini kan dijelaskan tentang bagaimana cara penulisan variabel yang benar di php sesuai dengan ketentuan dari PHP. berikut adalah cara penulisan variabel di php yang saya buat dalam beberapa point.

- Penulisan variabel PHP di awali dengan tanda \$.
- Variabel PHP bersifat Case Sensitive atau sensitif terhadap huruf besar dan kecil.
- Kemudian untuk mengisi suatu variabel bisa langsung menambahkan tanda sama dengan “=”.

Poin-poin di atas adalah cara penulisan dari variabel di PHP. penulisan variabel di php harus di awali dengan tanda dolar \$ dan kemudian di lanjutkan dengan nama dari variabel nya. misalnya :

```
<?php  
$nama;  
?>
```

Contoh di atas adalah salah satu contoh penulisan variabel di php. di awali dengan membuat tanda dolar \$ kemudian di lanjutkan dengan nama dari variabel tersebut. pada contoh di atas berarti variabel tersebut adalah variabel nama. jangan lupa untuk cara penulisan syntax php yang benar seperti yang sudah di jelaskan pada tutorial sebelumnya bahwa penulisan syntax php harus di awali dengan tag php pembuka( <?php ) dan di akhiri dengan tag php penutup ( ?> ). kemudian di akhir baris di tutup dengan tanda semicolon ( ; ).

Pada contoh kedua dijelaskan bahwa variabel di php memiliki sifat case sensitive berarti penulisan variabel php sangat peka terhadap huruf besar dan kecil. misalnya anda membuat variabe \$namaSaya, maka untuk memanggil variabel tersebut harus sesuai case sensitive nya seperti \$namaSaya.

Point ketiga mengatakan untuk mengisi suatu nilai atau informasi yang ingin di simpan kedalam variabel php bisa menggunakan tanda sama dengan " = ". Contoh nya

```
<?php  
$nama = "malasngoding";  
$umur = 28;  
?>
```

Contoh di atas adalah contoh pengisian data atau informasi ke dalam variabel PHP. perlu di perhatikan bahwa untuk mengisi data yang berupa string maka data harus di apit dengan petik ganda atau petik satu. tetapi jika nilai dari variabel tersebut berupa angka maka tidak perlu menggunakan tanda petik.

Karena pada tutorial sebelumnya anda sudah belajar tentang perintah “echo” di php yang berfungsi untuk menampilkan data maka pada tutorial ini kita akan menerapkannya juga sebagai contoh dari penggunaan variabel di PHP.

Buat sebuah file php dan simpan pada localhost (C://XAMPP/htdocs/). di sini file php tersebut saya beri nama *belajar.php*. kemudian isi dengan syntax berikut:

```
<?php  
$nama = "malasngoding";  
$umur = 28;  
echo $nama;
```

```
echo "<br/>";
echo $umur;
echo "<br/>";
echo "hai, perkenalkan nama saya ".$nama." dan saya berumur ".$umur;
?>
```

kemudian akses di browser dengan alamat <http://localhost/belajar.php>. dan jangan lupa mengaktifkan apache terlebih dahulu.



MALAS  
NGODING  
COM

mengenal variabel di PHP

Perhatikan pada contoh syntax di atas. tanda titik(.) digunakan untuk memisahkan string dan variabel. dan syntax <br/> adalah syntax break pada HTML.

## Mengupdate isi variabel

Variabel pada php juga bisa di update isinya. perhatikan contoh berikut:

```
<?php
$alamat = "jakarta";
$alamat = "bandung";
echo $alamat;
?>
```

dan hasilnya akan muncul bandung.



bandung

MALAS  
NGODING  
COM

#### Belajar PHP Mengenal Variabel Di PHP

Pada contoh update isi variabel di atas dapat di lihat bahwa pertama isi dari variabel alamat adalah jakarta. kemudian buat lagi variabel dengan isi yang berbeda, yaitu mengubah isi variabel menjadi bandung. maka hasilnya isi variabel alamat menjadi bandung.

## Belajar PHP Part 5 : Tipe Data Pada PHP

seperti bahasa pemrograman lainnya yang memiliki tipe data, sama halnya juga dengan bahasa pemrograman PHP. PHP memiliki beberapa tipe data. tipe data bisa dikatakan sebagai tipe dari isi variabel. ada yang berbentuk angka yang di sebut juga integer. dan yang berbentuk kalimat atau kata di sebut dengan string. berikut ini adalah beberapa tipe data pada PHP.

Tipe data pada PHP:

- String
- Integer
- Float
- Boolean
- Array
- Object
- NULL

dari tipe data PHP di atas hampir semua tipe data tersebut sering di gunakan dalam membangun sebuah aplikasi. berikut akan di jelaskan pengertian dan cara penggunaan masing-masing tipe data PHP di atas satu persatu.

### Tipe Data String pada PHP

Tipe data string adalah tipe data yang berbentuk text. dan untuk cara penulisan tipe data string di letakkan di tengah-tengah tanda petik. di awali dengan tanda petik dan di akhiri dengan tanda petik juga. berikut ini adalah contoh penulisan tipe data String di PHP.

```
<?php  
$tes = "Saya Pergi Kerja";  
// variabel tes di atas merupakan tipe data string karena berisi text atau  
kalimat.  
?>
```

Pada contoh tipe data string di atas kita misalkan terdapat sebuah variabel dengan nama tes yang berisi kalimat " Saya Pergi Kerja". maka variabel tes tersebut merupakan variabel yang bertipe data tes.

## Tipe Data Integer pada PHP

Tipe data integer adalah tipe data yang berbentuk angka yang berbentuk bilangan asli atau bilangan bulat. Untuk penulisan tipe data Integer tidak perlu menambahkan tanda petik karena akan di baca sebagai string jika anda menambahkan tanda petik pada tipe data integer. Berikut ini adalah contoh penulisan tipe data integer pada PHP.

```
<?php  
$bilangan_pertama = 12;  
$bilangan_kedua = 78;  
  
//Kedua variabel di atas merupakan variabel yang bertipe data integer.  
?>
```

Untuk cara penulisan variabel yang bertipe data integer dapat dilihat pada contoh tipe data integer di atas. Terdapat dua buah variabel yang bernama \$bilangan\_pertama dan \$bilangan\_kedua. Kedua variabel tersebut berisikan angka yang berbentuk bilangan bulat atau bilangan asli. Oleh sebab itu kedua variabel di atas disebut sebagai variabel yang bertipe data Integer.

## Tipe Data Float pada PHP

Tipe data float atau disebut juga tipe data double adalah tipe data yang berisi bilangan desimal. Cara penulisannya hampir sama dengan tipe data integer karena tidak memerlukan tanda petik di awal dan di akhir isi variabel float. Berikut ini adalah contoh penulisan variabel yang bertipe data Float pada PHP.

```
<?php  
$angka = 12.177;  
  
//variabel di atas adalah variabel yang bertipe data float karena berisi  
bilangan desimal.  
?>
```

## Tipe Data Boolean pada PHP

Tipe data boolean adalah tipe data yang hanya memiliki dua buah value atau isi. Yaitu true dan false. Atau 0 dan 1. true yang disebut dengan 1 dan false di sebut 0. Tipe data boolean sering digunakan untuk memeriksa ketersediaan atau memeriksa nilai kebenaran pada sebuah data. Yang akan kita pelajari lebih lanjut pada tutorial

belajar PHP selanjutnya di [malasngoding.com](http://malasngoding.com). berikut ini adalah cara penulisan variabel yang bertipe data boolean.

```
<?php
$x = false;
$y = true;
//variabel di atas bertipe data boolean karena berisi nilai benar atau salah.
?>
```

Pada contoh di atas kita misalkan terdapat sebuah variabel x yang berisi nilai false/salah dan variabel y yang berisi nilai true/benar.

## Tipe Data Array pada PHP

Array adalah sebuah tipe data yang menyimpan banyak isi di dalam sebuah variabel. jika kita ibaratkan array seperti sebuah tas yang di dalamnya bisa saja berisi pulpen, buku, penggaris dan lainnya. isi dari pada variabel array di tandai dengan masing-masing angka yang sudah di terapkan menurut urutannya. Belajar PHP Tipe Data Pada PHP. berikut adalah contoh penulisan variabel yang bertipe data array.

```
<?php
$anggota = array("Andi","Budi","Joni");
//variabel anggota di atas adalah variabel yang bertipe data array karena memiliki banyak isi pada.
?>
```

dapat di lihat pada contoh penulisan tipe data array di atas. untuk menggunakan tipe data array bisa menggunakan fungsi array(). kemudian di isi dengan isi dan di pisahkan dengan tanda koma pada tiap-tiap isi variabel. pada contoh variabel array anggota di atas terdapat andi, budi dan joni. dimana andi sebagai 0, budi sebagai 1, dan joni sebagai 2. angka yang diberikan pada tiap-tiap isi array di mulai dari 0. sekian tutorial belajar PHP dasar tentang pengenalan tipe data pada PHP.

## Belajar PHP Part 6 : Manipulasi String Pada PHP

Pada tutorial PHP ini kita akan belajar memanipulasi String dengan fungsi-fungsi yang sudah di sediakan oleh PHP. ada beberapa fungsi dasar untuk memanipulasi string dengan PHP yang harus anda ketahui. maksud dari manipulasi string di sini misalnya adalah anda dapat menghitung jumlah karakter yang terdapat pada string, memfilter karakter atau kata tertentu yang terdapat di dalam string, mengganti karakter tertentu pada string dengan karakter lain dan lainnya. berikut adalah beberapa fungsi php yang berfungsi untuk memanipulasi string.

### Menghitung jumlah karakter dengan strlen()

Untuk menghitung jumlah dari sebuah string anda dapat menggunakan fungsi `strlen()`. untuk contoh dan cara penulisan dari fungsi `strlen` untuk menghitung jumlah karakter di string php dapat di perhatikan pada contoh berikut ini:

```
<?php  
$kalimat = "Belajar pemrograman di malasngoding.com";  
echo strlen($kalimat);  
?>
```

dan jalankan pada browser



39

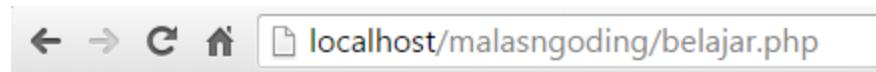
menghitung jumlah karakter pada string dengan `strlen` php

Pada contoh di atas dapat di lihat bahwa strin di masukkan ke dalam fungsi strlen(). dan kemudian di tampilkan dengan syntax echo yang menampilkan hasil dari strlen(). dan hasilnya pun muncul bahwa string tersebut berisi 39 karakter.

## Menghitung jumlah kata pada String dengan str\_word\_count()

Untuk menghitung jumlah kata yang terdapat pada sebuah string anda bisa menggunakan fungsi yang sudah di sediakan oleh php yaitu fungsi str\_word\_count(). berikut adalah cara penulisan fungsi str\_word\_count() untuk menghitung jumlah kata pada sebuah string.

```
<?php  
$kalimat = "Belajar pemrograman di malasngoding";  
echo str_word_count($kalimat);  
?>
```



4

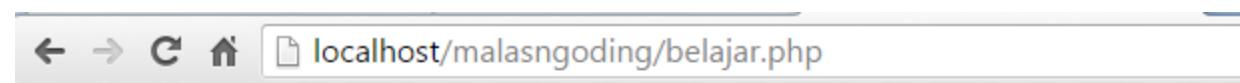
### Menghitung jumlah kata pada string dengan PHP

Variabel kalimat berisi sebuah string " Belajar Pemrograman di malasngoding". kemudian di gunakan fungsi php untuk menghitung kata pada string yaitu str\_word\_count() sehingga muncul hasil nya empat yang menandakan bahwa string tersebut memiliki empat buat kata.

## Membalik String Dengan strrev()

Fungsi strrev digunakan untuk membalikkan urutan karakter pada string. langsung saja untuk cara penulisan fungsi strrev() pada php seperti contoh berikut ini:

```
<?php  
$kalimat = "Belajar pemrograman di malasngoding";  
echo strrev($kalimat);  
?>
```



gnidognsalam id namargormep rajaleB

membalik urutan string dengan strrev pada php

dengan menggunakan fungsi strrev seperti contoh di atas berguna untuk membalikkan urutan string.

### Memfilter dan mengganti string dengan str\_replace()

Fungsi str\_replace berfungsi sebagai pengganti string. misalnya di dalam sebuah string terdapat kata atau hurus yang ingin di ganti dengan kata lain maka bisa menggunakan fungsi str\_replace(). untuk format penulisan nya seperti contoh berikut ini.

```
str_replace("yang ingin di ganti","pengganti","isi string");
```

dapat di lihat pada contoh penulisan fungsi str\_replace() php di atas bahwa fungsi str\_replace memiliki 3 parameter. parameter pertama adalah kata yang ingin di ganti, parameter kedua adalah kata yang digunakan sebagai pengganti, dan parameter ketiga adalah isi string. untuk contoh penggunaannya perhatikan pada contoh di bawah ini.

```
<?php  
$kalimat = "Belajar pemrograman di malasngoding";  
echo str_replace("Belajar","Tutorial",$kalimat);  
?>
```



## Tutorial pemrograman di malasngoding

### Belajar PHP Manipulasi String Pada PHP

dapat di lihat pada contoh di atas. awalnya isi dari variabel string kalimat adalah "Belajar Pemrograman di malasngoding". kemudian dengan fungsi str\_replace kita mengganti kata "belajar" dengan "tutorial". Belajar PHP Manipulasi String Pada PHP. sehingga hasil output dari str\_replace() adalah "tutorial pemrograman di malasngoding".

## Belajar PHP Part 7 : Kondisi IF ELSE Pada PHP

Kondisi If Else bisa anda ibaratkan dengan pengecekan jika maka pada kenyataannya. fungsi if else sangat di butuhkan pada saat anda ingin memeriksa sebuah kondisi. pada setiap logika terkadang kita memerlukan pengecekan suatu nilai yang jika terpenuhi untuk melakukan sesuatu dan jika pengecekan pertama tidak terpenuhi maka ada pengecekan kedua yaitu else. dapat kita bayangkan tentang kondisi if else ini seperti misalnya anda memiliki seorang teman. jika nama teman anda tersebut adalah andi maka dia adalah laki-laki jika tidak maka dia adalah perempuan.

Mungkin akan sedikit sulit jika di jelaskan. akan sangat membantu anda untuk memahami logika kondisi if else ini menggunakan contoh. berikut adalah contoh dari kondisi if else PHP.

### Mengenal Kondisi If-Else pada PHP

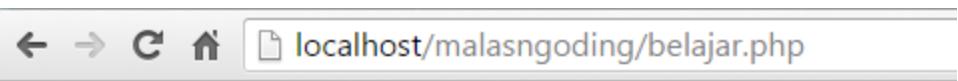
Kondisi If Else di perlukan untuk memeriksa suatu nilai untuk melakukan suatu aksi. berikut ini adalah format cara penulisan if else pada php.

```
<?php
if(kondisi){
    //jika kondisi terpenuhi
} else{
    //jika kondisi tidak terpenuhi
}
?>
```

dan berikut ini contoh sederhana dari penggunaan kondisi if else pada php. buat sebuah file php dengan nama belajar.php.

```
<?php
$teman = "andi";
if($teman == "andi"){
echo "dia adalah teman saya";
} else{
echo "dia bukan teman saya";
}
?>
```

Hasilnya



dia adalah teman saya

#### Pengenalan if else pada php

Perhatikan pada contoh di atas. di atas kita memiliki sebuah variabel yang bernama teman. kemudian isi dari variabel teman adalah "andi". dan kemudian dengan kondisi if else kita memeriksa variabel teman. cara pemahamannya begini "Jika variabel teman sama dengan andi maka tampilkan dia adalah teman saya, jika tidak tampilkan dia bukan teman saya".

dan kita coba jika variabel teman tidak sama dengan andi.

```
<?php
$teman = "andi";
if($teman == "Budi"){
echo "dia adalah teman saya";
}else{
echo "dia bukan teman saya";
}
?>
```

Perhatikan pada contoh di atas, variabel teman berisi "andi". dan kemudian kita melakukan pengecekan pada variabel teman. jika variabel teman sama dengan budi, maka tampilkan dia adalah teman saya. jika tidak tampilkan dia bukan teman saya. sedikit penjelasan lagi pada contoh di atas ini. sebagai contoh teman kita adalah si budi. jadi jika variabel teman sama dengan budi maka dia adalah teman saya. jika tidak maka dia bukan teman saya.



dia bukan teman saya

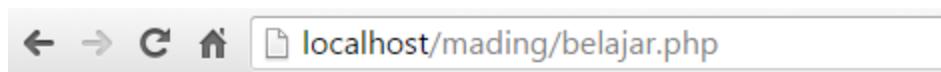
contoh penulisan kondisi if else pada php

## Mengenal Kondisi If-Elseif-Else

Elseif digunakan untuk memeriksa kondisi jika kondisi pertama(if) tidak terpenuhi. contohnya

```
<?php
$teman = "andi";
if($teman == "Budi"){
echo "Budi adalah teman saya";
}elseif($teman == "andi") {
echo "Andi adalah teman saya";
} else{
echo "Saya tidak punya teman";
}
?>
```

dan jika di jalankan



Andi adalah teman saya

#### Belajar PHP Kondisi IF ELSE Pada PHP

Variabel teman berisi "andi". jika variabel teman sama dengan "budi" maka tampilkan budi adalah teman saya. maka jika variabel teman sama dengan andi maka tampilkan andi adalah teman saya. jika tidak maka tampilkan saya tidak punya teman. jadi di sini kondisi "elseif" di gunakan jika anda membutuhkan pengecekan kondisi kedua atau seterusnya jika kondisi pertama tidak terpenuhi.

## Belajar PHP Part 8 : Mengenal Function Pada PHP

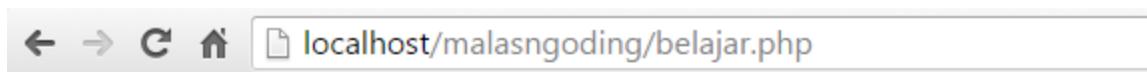
Function adalah kode program yang di buat khusus untuk mempersingkat sebuah aksi yang ingin di lakukan di dalam sebuah penulisan kode program. dan function dapat di panggil berkali-kali untuk melakukan tugas tertentu dalam penulisan kode program. function juga dapat kita buat sendiri untuk melakukan sebuah aksi.

pada contoh kehidupan nyata contohnya misalnya anda memerintahkan teman anda untuk menghidupkan lampu. jadi pada proses menghidupkan lampu oleh teman anda ini ada proses bangun dari tempat duduknya si teman anda ini kemudian dia menekan tombol saklar lampu untuk menghidupkan lampu. jadi kan akan sangat sia-sia jika anda ingin memerintahkan teman anda untuk menghidupkan lampu tetapi anda memerintahkannya secara berulang-ulang misalnya pertama anda menyuruhnya untuk bangun dari tempat duduknya kemudian menyuruhnya untuk menekan tombol saklar lampu untuk menghidupkan lampu. dan tentu akan hanya menyita waktu dan tenaga. oleh karena itu pasti anda hanya menyuruh tolong nyalakan lampu.

Jadi proses yang ada di dalam proses menyalakan lampu tadi tidak perlu lagi anda sebutkan karena sudah terdapat di dalam proses menghidupkan lampu. Belajar PHP Mengenal Function Pada PHP. agar lebih jelasnya coba anda perhatikan pada contoh berikut ini:

```
<?php
function tampilkan_nama() {
echo "Nama saya malas ngoding";
}

tampilkan_nama();
?>
```



## Nama saya malas ngoding

### pengertian dan cara penulisan function php

dari contoh di atas pasti anda sudah paham. jadi dengan function php ini anda dapat membuat sendiri suatu perintah yang dapat anda panggil berkali-kali. pada contoh ini saya membuat sebuah perintah dengan tampilkan\_nama. jadi isi dari perintah tampilkan\_nama ini adalah "echo 'nama saya malas ngoding'". sehingga pada saat perintah tampilkan\_nama di panggil maka akan di tampilkan 'nama saya malas ngoding'.

```
<?php  
function tampilan_nama(){  
    echo "Nama saya malas ngoding"; — Isi function  
}  
  
tampilan_nama();  
?>  
          \ Pemanggilan function
```

#### Belajar PHP Mengenal Function Pada PHP

Untuk cara penulisan function PHP harus di awali dengan syntax function yang menjelaskan bahwa yang kita buat adalah function. dan nama dari functionnya tidak boleh mengantung spasi. kemudian di tutup dengan tanda kurung pembuka dan kurung tutup " () ". dan isi dari perintah pada function terdapat di dalam tanda kurung kurawal pembuka " { " dan tanda kurung kurawal tutup " } ". cara pemanggilan function nya anda hanya perlu memanggil nama dari functionnya dan memberikan tanda buka dan tutup kurung " () " seperti contoh di atas.

## Belajar PHP Part 9 : Penggunaan Switch Case Pada PHP

Perintah Switch di gunakan untuk memeriksa kebenaran suatu nilai dengan memiliki banyak pemilihan atau nilai pengecekannya. hampir sama dengan penggunaan if else pada php yang telah di jelaskan pada tutorial belajar PHP dasar tentang mengenal kondisi if else pada php sebelumnya.

Pada tutorial ini akan di jelaskan tentang pengertian dan contoh penggunaan switch case PHP. langsung saja untuk contoh penulisan dan penggunaan Switch Case pada PHP .

```
<?php
$angka = 3;
switch ($angka) {
case 1:
    echo "Isi variabel angka adalah satu";
break;
case 2:
    echo "Isi variabel angka adalah dua";
break;
case 3:
    echo "Isi variabel angka adalah tiga";
break;
case 4:
    echo "Isi variabel angka adalah empat";
break;
case 5:
    echo "Isi variabel angka adalah lima";
break;
case 6:
    echo "Isi variabel angka adalah enam";
break;
default:
    echo "Isi variabel tidak di temukan";
break;
}
?>
```



## Isi variabel angka adalah tiga

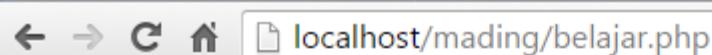
penulisan dan penggunaan switch case php

Perhatikan pada contoh di atas. terdapat sebuah variabel angka yang berisi angka 3. kemudian kita dapat memeriksa isi variabel angka tersebut untuk melakukan suatu perintah sesuai dengan isi variabel yang di temukan. pada parameter switch di atas kita mengisikan variabel angka.

Hal ini menjelaskan bahwa nilai variabel yang ingin di periksa adalah variabel angka. jika variabel angka sama dengan 1 maka di tampilkan "isi variabel angka adalah satu", jika variabel angka sama dengan 2 maka di tampilkan "isi variabel angka adalah dua", jika variabel angka sama dengan 3 maka di tampilkan "isi variabel angka adalah tiga", sampai seterusnya. jika suatu nilai di temukan maka akan di jalankan perintah nya masing-masing. perintah break di sini berfungsi untuk menghentikan pengecekan jika suatu pengecekan sudah terpenuhi kebutuhannya. jika seluruh pengecekan tidak terpenuhi atau tidak ada yang sesuai maka akan di jalankan nilai default yang menampilkan "isi variabel tidak di temukan".

```
<?php
$angka = 8;
switch ($angka) {
case 1:
    echo "Isi variabel angka adalah satu";
break;
case 2:
    echo "Isi variabel angka adalah dua";
break;
```

```
case 3:  
    echo "Isi variabel angka adalah tiga";  
break;  
case 4:  
    echo "Isi variabel angka adalah empat";  
break;  
case 5:  
    echo "Isi variabel angka adalah lima";  
break;  
case 6:  
    echo "Isi variabel angka adalah enam";  
break;  
default:  
    echo "Isi variabel tidak di temukan";  
break;  
}  
?>
```



**Isi variabel tidak di temukan**

### Belajar PHP Penggunaan Switch Case Pada PHP

contoh di atas adalah hasil jika pengecekan tidak di temukan nilainya.

## Belajar PHP Part 10 : Perulangan While Pada PHP

While adalah perintah pada PHP dan bahasa pemrograman lainnya untuk membuat sebuah perulangan yang tidak diketahui berapa kali perulangan tersebut dilakukan. Sering kali pada saat menuliskan kode program kita membutuhkan perintah perulangan while ini untuk membuat perulangan yang tidak perlu kita ketahui berapa kali perulangan harus dilakukan. misalnya seperti pada saat kita ingin menampilkan data dari database.

Untuk contoh penulisan dan struktur cara penggunaan perulangan while pada PHP dapat dilihat pada contoh di bawah ini

```
while( Kondisi ){
    Statement.
}
```

Dapat dilihat pada contoh penulisan while di atas setelah menuliskan while kemudian di dalam tanda kurung berisi kondisi nilai yang harus terpenuhi agar while dijalankan. kemudian di dalam kurung kurawal berisi statement yang di jalankan dalam bentuk perulangan apabila kondisi terpenuhi. statement di sini maksudnya adalah perintah syntax yang akan di jalankan dalam bentuk perulangan apabila kondisi terpenuhi.

```
<?php
$x = 1;

while($x <= 10) {
    echo "Angka $x <br>";
    $x++;
}
?>
```



Angka 1  
Angka 2  
Angka 3  
Angka 4  
Angka 5  
Angka 6  
Angka 7  
Angka 8  
Angka 9  
Angka 10

#### contoh penulisan while php

Perhatikan pada contoh di atas. perintah while di gunakan untuk melakukan perulangan. variabel x memiliki nilai 10. kemudian pada kondisi while kita memberikan nilai variabel x lebih kecil dan sama dengan 10. jadi hasil yang muncul adalah 1 sampai 10. perintah ++ pada variabel x untuk memberitahukan perulangan untuk melakukan perulangan terus menerus sampil menambah nilai awal variabel x secara terus menerus. sekian tutorial php dasar pengertian dan contoh penggunaan while di php.

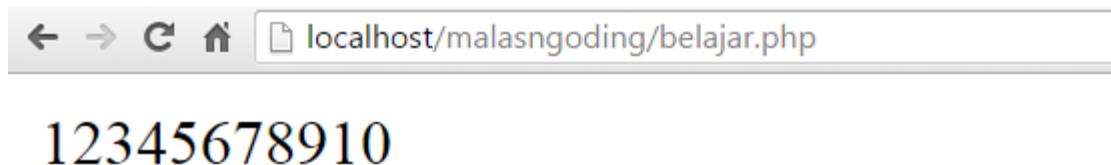
Sekianlah sekilas penjelasan tentang Belajar PHP Perulangan While Pada PHP semoga dapat bermanfaat. akan di lanjutkan pada tutorial selanjutnya di edisi belajar PHP dasar ini. selanjutnya akan di bahas tentang perulangan for di php.

## Belajar PHP Part 11 : Perulangan For Pada PHP

Perulangan For digunakan untuk membuat sebuah perulangan pada pemrograman termasuk dalam bahasa pemrograman PHP. tujuannya hampir sama dengan perulangan while. hanya saja perulangan while memiliki jumlah batas yang tidak di ketahui. sedangkan perulangan for memiliki batas berapa kali perulangan di ulangi dan batas dari perulangannya sudah di tentukan pada kondisi perulangan for.

untuk pemahaman yang lebih jelasnya bisa di lihat pada contoh perulangan for berikut ini.

```
<?php
for ($x=1;$x<=10;$x++) {
echo $x;
//membuat perulangan yang menampilkan angka satu sampai sepuluh sesuai dengan
aturan yang sudah di buat pada kondisi di atas.
}
?>
```



### Belajar PHP Perulangan For Pada PHP

Pada contoh perulangan for pada php di atas kita menentukan nilai dari variabel x sama dengan 1, kemudian variabel lebih kecil dan sama dengan 10. artinya bilangan 1 sampai dengan 10. kemudian menampilkan isi dari variabel dalam bentuk

perulangan for dengan menampilkannya di dalam perulangan for. untuk penulisan perulangan for kondisi perulangan di letakkan di dalam tanda kurung. dan aksi dari perulangannya di letakkan di dalam tanda kurung kurawal ( {} ).

Kita juga bisa membuat perulangan yang menampilkan string dengan jumlah yang kita inginkan. teknik ini sangat kita butuhkan pada saat membuat program. misalnya anda ingin menampilkan sebuah kalimat sebanyak 20 kali. tentu akan sia-sia dan buang-buang waktu jika kita copy paste sebanyak 20 kali. akan lebih mudah jika kita buat menggunakan perulangan for.

```
<?php  
for ($kata=1;$kata<=20;$kata++) {  
    echo "Belajar pemrograman PHP di www.malasngoding.com". "<br/>";  
}  
?>
```

localhost/malasngoding/belajar.php

Belajar pemrograman PHP di www.malasngoding.com  
Belajar pemrograman PHP di www.malasngoding.com

Pengertian dan cara penggunaan perulangan for php

Sekianlah tutorial php dasar perulangan for pada php.

## Belajar PHP Part 12 : Mengenal Array Pada PHP

Array adalah suatu tipe data variabel yang mampu menyimpan banyak data atau nilai. tipe data array sudah di jelaskan pada tutorial sebelumnya pada sub judul mengenal tipe data array pada php.

Karena array memiliki banyak data, maka isi dari array di tandai dengan angka pada tiap-tiap isi dari array. angka pada masing-masing isi array di mulai dari 0,1,2,3 sampai seterusnya. Ada dua cara penamaan atau pemanggilan isi array. yaitu dengan cara memanggil menggunakan angka default array, dan bisa juga dengan memberikan penamaan pada masing-masing isi array.

### Membuat dan memanggil data pada array secara biasa

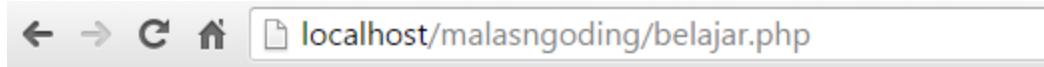
Secara default, tiap-tiap isi dari pada variabel array di beri nomor urut, yaitu di mulai dari 0. seperti pada contoh berikut ini.

```
<?php
//membuat array yang berisi nama buah-buahan
$buah = array('semangka','jeruk','apel','anggur');
//menampilkan data array dengan nomor urut 2
echo $buah[2];
?>
```

Kita memiliki sebuah variabel array dengan nama variabel buah. dan di dalamnya terdapat nama buah semangka, jeruk, apel, dan anggur. berarti penomoran dari isi array ini semangka memiliki nomor urut 0, jeruk memiliki nomor urut 1, apel memiliki nomor urut 2, dan anggur memiliki nomor urut 3. jadi untuk memanggil atau menampilkan buah apel kita menggunakan syntax:

```
echo $buah[2];
```

karena buah apel memiliki nomor urut 2. dan jika di jalankan pada browser maka hasilnya akan muncul seperti gambar di bawah ini.



# apel

contoh pemanggilan array dengan nomor urut

Membuat dan memberi penamaan pada isi array

Untuk memberikan nama pada tiap-tiap data array cara nya:

```
<?php  
  
//penamaan isi array  
$buah['semangka'] = "isinya merah";  
$buah['jeruk'] = "rasanya manis";  
$buah['apel'] = "warnanya merah";  
$buah['anggur'] = "harganya mahal";  
  
// menampilkan isi array yang bernama jeruk  
echo $buah['jeruk'];  
  
?>
```



# rasanya manis

## Belajar PHP Mengenal Array Pada PHP

Perhatikan pada contoh penamaan array pada contoh di atas. bahwa kita membuat variabel array dengan nama buah dan semuanya memiliki isi.

Atau bisa juga dengan menggunakan syntax seperti berikut untuk memberi penamaan pada variabel array.

```
<?php  
  
//penamaan isi array variabel buah  
$buah = array(  
    'semangka' => "isinya merah",  
    'jeruk' => "rasanya manis",  
    'apel' => "warnanya merah",  
    'anggur' => "harganya mahal"  
);  
  
// menampilkan isi array yang bernama jeruk  
echo $buah['jeruk'];  
  
?>
```

Seperti contoh di atas kita secara langsung memberikan nama pada isi dari variabel array. variabel array semangka berisi isinya merah, variabel array jeruk berisi

rasanya manis, variabel array apel berisi warnanya merah, variabel array anggur berisi harganya mahal. karena di sini kita memanggil variabel array jeruk maka yang tampil adalah rasanya manis.



# rasanya manis

pemberian nama pada isi array di php

## Contoh Menerapkan Array dengan For

```
<?php  
  
//membuat array yang berisi nama buah-buahan  
$buah = array('semangka','jeruk','apel','anggur');  
  
//count() untuk menghitung isi array.  
for($x=0;$x<count($buah);$x++) {  
    echo $buah[$x]."<br/>";  
}  
?>
```

semangka  
jeruk  
apel  
anggur

#### contoh array dan for pada php

function count() untuk menghitung jumlah isi array. perulangan for di atas akan menghasilkan 0 sampai 3. kenapa tiga? karena kita menggunakan lebih kecil dari jumlah isi array. sehingga kita bisa menampilkan isi array menggunakan urutan angka for.

## Belajar PHP Part 13 : Menampilkan Tanggal Pada PHP

Bahasa pemrograman PHP di lengkapi dengan berbagai fungsi yang sangat memudahkan dan membantu para programmer dalam membuat sebuah program. salah satunya adalah fungsi yang digunakan untuk menampilkan tanggal. untuk menampilkan tanggal pada PHP bisa menggunakan fungsi date().

Dan format tanggal bisa di atur di dalam tanda kurung pada fungsi date(). data tanggal yang di ambil adalah tanggal dari server. biasanya pada saat membuat website atau sistem informasi. fungsi date ini di gunakan untuk menampilkan tanggal, mengambil dan menyimpan data tanggal. mungkin saja misalnya kita membuat sebuah aplikasi yang juga dapat menyimpan tanggal pendaftaran, tanggal penginputan data, tanggal jadwal dan lainnya.

Semua aplikasi atau website besar juga pasti menggunakan fungsi date() untuk mendapatkan data tanggal. termasuk juga CMS WordPress yang menggunakan fungsi date ini untuk menyimpan data tanggal sebuah postingan. oleh sebab itu pada tutorial belajar PHP cara menampilkan tanggal di php ini akan di jelaskan tentang cara menampilkan tanggal dengan fungsi php. dan mengatur output tanggal yang ingin di tampilkan sesuai dengan keinginan seperti format tanggal, bulan dan tahun, baik itu menampilkan nya dalam bentuk huruf, angka, bentuk 2 digit, atau 4 digit. berikut ini adalah cara menampilkan tanggal pada php.

Adapun bentuk format penggunaan dan penulisan fungsi date di php adalah:

```
date(format,timestamp);
```

pada parameter format di atas di isi dengan format penulisan tanggal yang ingin di tampilkan. parameter ini harus di isi untuk membuat format tanggal. dan parameter timestamp merupakan parameter optional artinya boleh di isi boleh juga di kosongkan.

### Cara menampilkan tanggal di PHP

Berikut ini adalah beberapa format karakter tanggal yang dapat digunakan sesuai keinginan. masing-masing format karakter ini memiliki kegunaannya masing-masing:

- l = karakter L (huruf kecil) di gunakan untuk menampilkan nama hari dalam bentuk huruf.
- D = karakter D (huruf besar) digunakan untuk menampilkan nama hari hari ini dalam bentuk huruf tetapi dalam bentuk tiga digit.
- d = karakter d digunakan untuk menampilkan tanggal dalam bentuk angka yaitu dari 1 sampai dengan 31.
- M = karakter m digunakan untuk format menampilkan bulan sekarang dalam bentuk angka yaitu misal dari angka bulan 1 sampai angka bulan 12.
- m = karakter m(huruf kecil) digunakan untuk menampilkan nama bulan sekarang. nama bulan di tampilkan dalam bentuk huruf.
- Y = karakter Y ( huruf besar ) digunakan untuk menampilkan tahun dalam bentuk angka empat digit.
- y = karakter y (huruf kecil) digunakan untuk menampilkan angka tahun dalam bentuk dua digit. angka yang di tampilkan adalah dua angka terakhir dari tahun sekarang.

Kemudian terdapat beberapa karakter tambahan yang bisa anda tambahkan untuk membuat format tanggal sesuai dengan keinginan anda seperti menambahkan karakter "/"(garis miring), "."(titik), "-"(strip). ","(koma) dan karakter lainnya sesuka anda.

```
<?php
echo date('l, d-m-Y');
echo "<br/>";
echo date('d / M / y');
echo "<br/>";
echo date('D - M / Y');
?>
```



Tuesday, 29-12-2015

29 / Dec / 15

Tue - Dec / 2015

menampilkan tanggal di php

## Menampilkan waktu di PHP

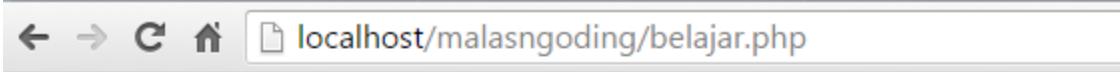
Dalam penggunaan fungsi date() kita juga dapat menampilkan waktu seperti jam, menit,detik serta am dan pm. untuk format karakter untuk menampilkan waktu jam bisa menggunakan:

- s = untuk menampilkan detik saat ini. format yang di tampilkan angka dari 0 sampai dengan 59.
- i = untuk menampilkan menit sekarang dalam format 0 sampai dengan 59.
- h = untuk menampilkan jam sekarang salam format 1 sampai 12.
- H = H(huruf besar) digunakan untuk menampilkan format jam dalam bentuk 24 jam.
- a = untuk menampilkan waktu am atau pm sekarang.

sama seperti format penulisan tanggal, anda dapat menambahkan karakter-karakter sesuai dengan keinginan. untuk contoh menampilkan waktu jam pada php silahkan perhatikan contoh berikut ini.

```
<?php
```

```
// menampilkan jam sekarang  
echo date('H:i:s a');  
echo "<br/>";  
  
//kombinasi format tanggal dan jam  
echo date('l, d-m-Y h:i:s a');  
?>
```



14:06:26 pm  
Tuesday, 29-12-2015 02:06:26 pm

Belajar PHP Menampilkan Tanggal Pada PHP

## Belajar PHP Part 14 : Include Dan Require Pada PHP

Sesuai dengan judul pada tutorial belajar PHP dasar Part 14 di ini kita akan membahas tentang Mengenal Include Dan Require Pada PHP. pada tutorial belajar php dasar ini akan di jelaskan apa itu fungsi include(), require() ? apa kegunaan dari fungsi include() dan require() ? kapan kita harus menggunakan fungsi include() dan require() ? dan apa perbedaan dari masing-masing fungsi include() dan require() di PHP? dan kenapa menggunakan include\_once dan require\_once? berikut adalah pengertian dari fungsi include(), require() dan require\_once() pada PHP.

Fungsi include() dan require() adalah suatu perintah atau fungsi di dalam pemrograman PHP untuk menyisipkan sebuah file PHP kedalam file PHP yang lainnya. dengan menggunakan teknik penyisipan file php menggunakan fungsi include(), require() dan require\_once() akan sangat membantu kita dalam membuat program karena kita tidak perlu menuliskan kode program secara berulang-ulang. kita hanya tinggal menyisipkan file php tersebut kedalam file php lainnya sesuai dengan kebutuhan.

### Cara penulisan dan penggunaan Include dan Require

Untuk cara penulisan dan penggunaan fungsi include() dan require() caranya sangat mudah. di contohkan di sini misalnya kita memiliki file index.php. dan kita ingin menyisipkan file header.php di pada bagian atas index.php. maka caranya:

```
include('header.php');
```

contoh di atas adalah cara penulisan include file header.php ke file index.php. dan jika menggunakan require().

```
require('header.php');
```

pada parameter include atau require isikan dengan nama file php yang ingin di sisipkan. contoh di atas digunakan jika file yang ingin di sisipkan terletak pada satu

folder atau satu direktori. bagaimana jika file yang ingin di sisipkan terletak di luar folder. maka bisa menambahkan “..”. contohnya:

```
include('..../header.php');
```

Atau

```
require('..../header.php');
```

dan jika file yang ingin di sisipkan terletak pada folder lain bisa menambahkan nama folder tersebut.

```
include('namafolder/header.php');
```

## Contoh penggunaan fungsi include dan require PHP

Untuk contoh penggunaan dari fungsi include dan require pada PHP . siapkan dua buah file PHP. untuk contoh di sini file yang pertama saya buat file index.php. kemudian file yang kedua tes.php

isi file index.php

```
<?php
//menyisipkan file tes.php di sini
include('tes.php');

//syntax di bawah adalah isi dari file index.php
echo "Belajar Include() dan Require() di www.malasngoding.com";
?>
```

isi file tes.php

```
<?php
echo "<h1> ISI FILE TES.PHP </h1>";
?>
```

# ISI FILE TES.PHP

Belajar Include() dan Require() di [www.malasngoding.com](http://www.malasngoding.com)

## Belajar PHP Mengenal Include Dan Require Pada PHP

Bisa di perhatikan pada contoh di atas bahwa isi dari file tes.php juga di tampilkan pada saat file index.php di jalankan. contoh di atas adalah contoh penggunaan dari fungsi include pada php. anda bisa mencobanya dengan menggunakan fungsi require(). untuk perbedaan dari fungsi include() dan require() adalah jika file yang di sisipkan dengan menggunakan fungsi include tidak tersedia atau salah dalam peletakan lokasi maka file akan tetap di jalankan dengan mengabaikan error. tetapi jika menggunakan require() dan file yang di sisipkan tidak tersedia atau salah dalam peletakan lokasi maka isi dari file tidak akan di lanjutkan dan akan di hentikan penyisipannya pada letak error.

## Perbedaan Include, Require dan Include\_once, require\_once

Dan bagaimana dengan perbedaan dari fungsi include(), require() tanpa menggunakan "once". dengan penggunaan include\_once() atau require\_once() maka berarti penyisipan hanya di panggil sekali saja. walaupun di sisipkan beberapa kali di dalam sebuah file maka dengan menggunakan fungsi include\_once() atau require\_once() penyisipan tetap di panggil sekali saja. beda dengan include() dan require() yang jika di sisipkan beberapa kali pada sebuah file maka akan menyebabkan

error atau redeclare(deklarasi ganda). untuk cara penulisan include\_once() dan require\_once() juga sama seperti penulisan include() dan require().

```
include_once('tes.php');
```

Atau

```
require_once('tes.php');
```

## Belajar PHP Part 15 : Penanganan Form Pada PHP

Pada tutorial Belajar PHP Part 15 : Penanganan Form Pada PHP ini akan di jelaskan tentang bagaimana cara menangkap data yang di input atau mengolah data inputan pada form. ada dua metode yang di gunakan untuk mengolah form inputan. yaitu GET dan POST. bagaimana cara melakukan penanganan form atau form handling pada PHP dengan menggunakan metode GET dan POST ? dan bagaimana cara menangkap data yang di inputkan pada form dengan menggunakan perintah \$\_GET, \$\_POST dan \$\_REQUEST. perhatikan penjelasan berikut ini.

Untuk melakukan penanganan form atau form handling, kita harus menentukan terlebih dahulu form yang kita buat menggunakan metode apa? metode get atau post. berikut ini akan di jelaskan cara penanganan form yang menggunakan metode get atau metode post dengan php.

untuk contoh penggunaan GET dan POST pada PHP silahkan sediakan dua buah file php. yaitu index.php dan tampil.php. jadi pada contoh ini kita akan membuat sebuah form penginputan nama. dan apabila form tersebut di submit maka akan muncul nama yang di inputkan pada form. pada file index.php kita jadikan sebagai form dan pada file tampil.php kita jadikan sebagai file php yang bertugas untuk menangkap data inputan kemudian menampilkannya. contoh penggunaan method GET pada form HTML dan PHP silahkan perhatikan pada contoh membuat penanganan form dengan method GET berikut ini.

### Membuat penanganan form dengan method GET

berikut ini adalah contoh penanganan form dengan method GET

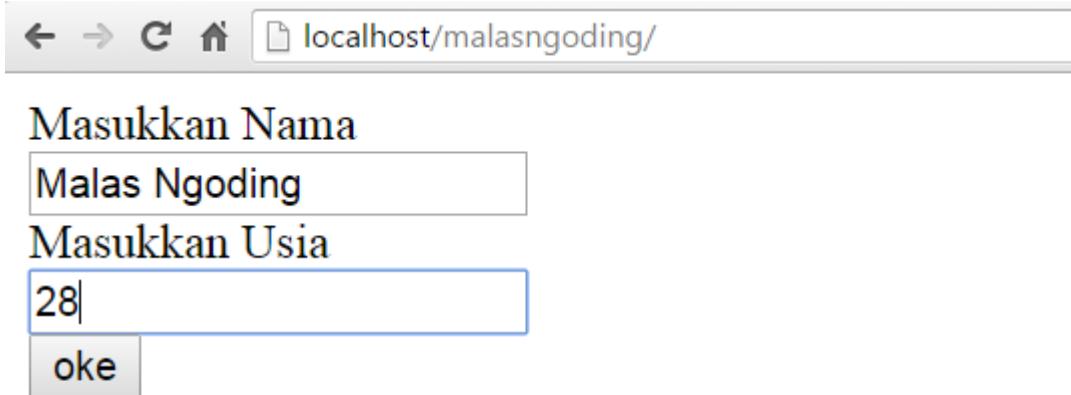
index.php

```
<!-- penanganan form dengan method GET -->
<form method="get" action="tampil.php">
    <label>Masukkan Nama</label><br/>
    <input type="text" name="nama"><br/>
    <label>Masukkan Usia</label><br/>
    <input type="text" name="usia"><br/>
    <input type="submit" value="oke">
</form>
```

## tampil.php

```
<?php  
// menangkap data nama dengan method nama  
$nama = $_GET['nama'];  
// menangkap data usia dengan method nama  
$usia = $_GET['usia'];  
  
// menampilkan data nama  
echo "Nama anda adalah " . $nama;  
echo "<br/>";  
// menampilkan data usia  
echo "Usia anda adalah " . $usia;  
?>
```

kemudian jalankan index.php



The screenshot shows a web browser window with the URL `localhost/malasngoding/` in the address bar. Below the address bar is a form with two text input fields and one button. The first field is labeled **Masukkan Nama** and contains the text **Malas Ngoding**. The second field is labeled **Masukkan Usia** and contains the text **28**. Below these fields is a button labeled **oke**.

contoh form get

dan output yang dihasilkan



Nama anda adalah Malas Ngoding  
Usia anda adalah 28

Data di kirim melalui  
URL

### Hasil Output

dapat di lihat pada contoh di atas, untuk menggunakan method get pada form tuliskan get pada atribut method pada tag form pembuka. dan pada atribut action tuliskan nama file php yang bertugas untuk mengelola atau menangkap data dari form tersebut.

```
<form method="get" action="tampil.php">
```

dan pada hasil output dapat di lihat bahwa data yang di kirimkan oleh form get melalui url seperti pada gambar hasil output di atas.

untuk penangkapan data pada file tampil.php karena form menggunakan method "get" maka pada saat penangkapan data get gunakan `$_GET` seperti pada file tampil.php

```
<?php
// menangkap data nama dengan method get
$nama = $_GET['nama'];
// menangkap data usia dengan method get
$usia = $_GET['usia'];

// menampilkan data nama
echo "Nama anda adalah " . $nama;
echo "<br/>";
// menampilkan data usia
echo "Usia anda adalah " . $usia;
?>
```

Akan di jelaskan sedikit untuk cara penangkapan data yang menggunakan method get. pada contoh di atas kita memasukkan data yang di tangkap menggunakan get kedalam variabel nama. begitu juga dengan data usia. data yang di tangkap sesuai dengan nama form . misalnya pada form input nama tadi form kita beri nama degan "nama". dan form usia kita beri nama dengan "usia".

```
<input type="text" name="nama"><br/>
<input type="text" name="usia"><br/>
```

perhatikan

```
// menangkap data nama dengan method get
$nama = $_GET['nama'];
// menangkap data usia dengan method get
$usia = $_GET['usia'];
```

Atau

```
$ GET['nama form'];
```

## Membuat penanganan form dengan method POST

Membuat penanganan form dengan method post sama saja dengan get. hanya saja pada method form gunakan post bukan get. dan untuk cara menangkap data yang di kirim dari form yang ber method post gunakan `$_POST`. contoh penggunaan method post pada php.

index.php

```
<!-- penanganan form dengan method POST -->
<form method="post" action="tampil.php">
    <label>Masukkan Nama</label><br/>
    <input type="text" name="nama"><br/>
    <label>Masukkan Usia</label><br/>
    <input type="text" name="usia"><br/>
    <input type="submit" value="oke">
</form>
```

tampil.php

```
<?php
// menangkap data nama dengan method post
$nama = $_POST['nama'];
// menangkap data usia dengan method post
$usia = $_POST['usia'];

// menampilkan data nama
```

```
echo "Nama anda adalah " . $nama;  
echo "<br/>";  
// menampilkan data usia  
echo "Usia anda adalah " . $usia;  
?>
```

The screenshot shows a web browser window with the URL `localhost/malasngoding/` in the address bar. The page displays a form with two input fields and one button. The first input field contains the text `Malas Ngoding`. The second input field contains the text `28`. Below the inputs is a button labeled `oke`.

contoh form post

hasilnya



Nama anda adalah Malas Ngoding  
Usia anda adalah 28

#### Belajar PHP Penanganan Form Pada PHP

## Membuat penanganan form dengan \$\_REQUEST

`$_GET` digunakan untuk menangkap data dari form get. `$_POST` digunakan untuk menangkap data dari form post. dan `$_REQUEST` bisa digunakan untuk menangkap data dari method get dan method post. Belajar PHP Penanganan Form Pada PHP

## Kelebihan dan Kekurangan GET dan POST

Method get sangat tidak di sarankan jika digunakan untuk membuat login karena akan sangat tidak lucu jika username dan password yang di inputkan akan terlihat pada url browser.

Method get lebih mudah di gunakan untuk pengambilan atau penyimpanan data sementara yang di butuhkan dalam logika pemrograman. karena lebih praktis dalam menyimpan data yang bersifat sementara.

Method post digunakan untuk membuat login karena membuat data yang di inputkan tidak terlihat.

## Belajar PHP Part 16 : Operator Aritmatika di PHP

Operator aritmatika atau disebut juga operator matematika merupakan operator yang digunakan untuk pengolahan bilangan. contohnya seperti pembagian, perkalian, penjumlahan, pengurangan, modulus, min dan plus. di dalam bahasa pemrograman PHP mempunyai beberapa operator aritmatika yaitu:

- Penjumlahan  
operator penjumlahan di php digunakan untuk menjumlahkan bilangan atau variabel. contohnya `$a + $b`.
- Pengurangan  
Operator pengurangan di php digunakan untuk mengurangkan bilangan atau variabel. contohnya `$a - $b`.
- Perkalian  
Operator perkalian di php digunakan untuk mengalikan bilangan atau variabel. untuk membuat perkalian di php gunakan tanda bintang `" * "`. contohnya `$a * $b`.
- Pembagian  
Operator pembagian di php digunakan untuk membuat pembagian bilangan atau variabel. untuk membuat pembagian di php gunakan garis miring `" / "`. contohnya `$a / $b`.
- Minus  
Contoh membuat operator minus di php misalnya seperti `" -$a "`.
- Plus  
Contoh membuat operator plus di php misalnya seperti `" +$a "`.
- Modulus  
Operator modulus adalah operator untuk menemukan hasil dari pembagian. untuk penulisan operator modulus di php menggunakan tanda persen `" % "`. contohnya `" $a % $b "`. operator modulus di gunakan untuk mendapatkan hasil sisa dari pembagian. misalnya  $18 \% 8$  maka hasilnya adalah 2. karena  
 $18 / 8 = 2$   
 $8 \times 2 = 16$   
 $18 - 16 = 2$

## Cara membuat operator aritmatika di PHP

Berikut adalah contoh penulisan dari operator aritmatika yang sudah di jelaskan di atas. berikut adalah membuat operator penjumlahan di php.

```
<?php  
$a = 4;  
$b = 19;  
//menjumlahkan variabel a dengan variabel b  
echo $a + $b;  
?>
```

## Contoh operator pengurangan di PHP

```
<?php  
$a = 4;  
$b = 19;  
//Operator pengurangan variabel a dengan variabel b  
echo $a - $b;  
?>
```

## Contoh operator perkalian di PHP

```
<?php  
$a = 4;  
$b = 19;  
//perkalian variabel a dengan variabel b  
echo $a * $b;  
?>
```

## contoh operator pembagian di PHP

```
<?php  
$a = 4;  
$b = 19;  
//pembagian variabel a dengan variabel b  
echo $a / $b;  
?>
```

## contoh operator modulus di PHP

```
<?php  
$a = 18;  
$b = 8;  
//menjumlahkan variabel a dengan variabel b  
echo $a % $b;
```

Pada contoh di atas tersedia dua variabel yaitu variabel a dan variabel b. variabel a dan variabel b sudah berisi nilai masing-masing dan kemudian untuk melakukan operator aritmatika atau pengolahan kedua bilangan tersebut bisa menggunakan operatornya masing-masing secara langsung seperti yang sudah di jelaskan pada penjelasan operator aritmatika di php.

# Belajar PHP Part 17 : Pengertian dan Pengenalan Foreach

Sebelumnya di kategori tutorial php kita telah belajar tentang perulangan dasar dengan menggunakan for, do while, while. maka pada tutorial ini akan saya jelaskan satu fungsi lagi di dalam PHP. yang bisa di gunakan untuk membuat perulangan. tetapi bukan perulangan biasa. namun perulangan untuk array. yaitu menggunakan foreach.

Foreach di gunakan untuk perulangan yang datanya dalam bentuk array. terkadang kita memiliki data dalam bentuk array yang harus kita tampilkan dengan menggunakan perulangan for. dengan membuat variabel counter. tetapi dengan menggunakan foreach kita tidak perlu lagi menampilkan data dari array dengan variabel counter seperti di perulangan for.

Nanti jika anda mulai masuk ke framework codeigniter, teman-teman akan sangat membutuhkan foreach dalam menampilkan data. baca di sini tutorial codeigniter di [www.malasngoding.com](http://www.malasngoding.com).

## Cara Penggunaan Foreach di PHP

Sebelumnya anda telah mempelajari tentang perulangan dengan menggunakan for dan while. dan contoh cara menampilkan data array dengan menggunakan for adalah sebagai berikut.

```
<?php  
echo "<h1>WWW.MALASNGODING.COM</h1>";  
  
$data = array("Andi", "Budi", "Caca");  
  
for($x=0;$x<count($data);$x++) {  
echo $data[$x]."<br/>";  
}  
?>
```



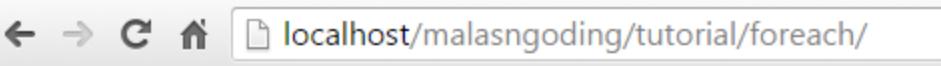
pengertian dan cara menggunakan foreach pada php

Baca selengkapnya tentang pengertian dan cara menggunakan perulangan for. Dan bagaimana jika menggunakan foreach untuk menampilkan data yang berbentuk array ?

```
<?php
echo "<h1>WWW.MALASNGODING.COM</h1>";

$data = array("Andi", "Budi", "Caca");

foreach($data as $d) {
    echo $d."<br/>";
}
?>
```



belajar php pengertian dan pengenalan foreach pada php

Jika menggunakan perulangan for untuk menampilkan data array, maka kita membutuhkan variabel counter untuk menampilkan data nya berurutan dengan angka

yang di hasilkan oleh perulangan for. seperti contoh yang dapat teman-teman perhatikan pada contoh di atas.

Dan jika menggunakan foreach. kita tidak perlu lagi menggunakan variabel counter untuk menampilkan data yang berbentuk array.

## Belajar PHP Part 18 : Explode Dan Implode Pada PHP

Kali ini kita akan membahas dan belajar tentang pengertian dari fungsi explode() dan implode() pada php. pasti teman-teman mungkin sudah ada yang pernah mendengar dan bahkan sudah pernah menggunakan. mungkin ada juga teman-teman yang belum mengetahui apa sih kegunaan dari kedua fungsi php ini. yaitu fungsi explode dan implode. Langsung kita masuk ke pembahasan saja agar tidak bingung.

Explode dan implode adalah fungsi yang di sediakan oleh php untuk memisahkan atau menggabungkan string. string yang di pisahkan akan menjadi bentuk array yang memiliki penomoran nya masing-masing. lebih detailnya silahkan teman-teman perhatikan penjelasan berikut ini.

- Explode. fungsi explode adalah fungsi php yang memiliki fungsi sebagai pemisah string. jadi sebuah string akan di pecahkan menurut tanda pisah yang di tentukan sendiri. jadi string yang telah di pecah, akan menjadi array. cara penulisannya : explode().
- Implode. fungsi implode adalah fungsi php yang memiliki kegunaan untuk menggabungkan kembali string yang telah di pecahkan tadi menggunakan fungsi explode. cara penulisannya : implode().

### Memecahkan string dengan fungsi explode()

Seperti yang sudah di jelaskan di atas tadi. explode berguna untuk memecahkan string menjadi array. berikut adalah contoh.

```
<?php  
echo "<h2>www.malasngoding.com</h2>";  
echo "<br/>";  
  
$kalimat = "belajar pemrogaman di malas ngoding";  
  
// memisahkan string menjadi array  
$data = explode(" ", $kalimat);  
  
// string menjadi array  
print_r($data);  
//print_r berguna untuk menampilkan isi array  
  
echo "<br/>";
```

```
echo "<br/>";  
  
echo " Data 1 = " . $data[0];  
echo "<br/>";  
echo " Data 2 = " . $data[1];  
echo "<br/>";  
echo " Data 3 = " . $data[2];  
echo "<br/>";  
echo " Data 4 = " . $data[3];  
echo "<br/>";  
echo " Data 5 = " . $data[4];  
  
?>
```

coba teman-teaman perhatikan. di sini kita memiliki sebuah string di dalam variabel kalimat.

```
$kalimat = "belajar pemrogaman di malas ngoding";
```

dan kemudian kita pecahkan string ini menjadi array dengan menggunakan fungsi explode().

```
// memisahkan string menjadi array  
$data = explode(" ", $kalimat);
```

Pada fungsi explode isikan pemisah nya. di sini saya mengisi dengan spasi. jadi hasilnya string akan di pisahkan menurut/setiap tanda spasi.

Sampai di sini, variabel data sudah berisi data string yang sudah menjadi array. untuk melihat isi string nya.

```
//string menjadi array  
print_r($data);  
//print_r berguna untuk menampilkan isi array
```

dan untuk menampilkan data nya satu persatu.

```
echo " Data 1 = " . $data[0];  
echo "<br/>";  
echo " Data 2 = " . $data[1];  
echo "<br/>";  
echo " Data 3 = " . $data[2];  
echo "<br/>";  
echo " Data 4 = " . $data[3];  
echo "<br/>";  
echo " Data 5 = " . $data[4];
```



## www.malasngoding.com

```
Array ( [0] => belajar [1] => pemrograman [2] => di [3] => malas [4] => ngoding )
```

```
Data 1 = belajar  
Data 2 = pemrograman  
Data 3 = di  
Data 4 = malas  
Data 5 = ngoding
```

pengertian dan contoh penggunaan fungsi explode php

## Menggabungkan String dengan implode()

Jika explode adalah pemecah string, maka implode adalah penggabung kembali string yang telah dipecahkan oleh explode.

```
<?php  
  
echo "<h2>www.malasngoding.com</h2>";  
echo "<br/>";  
  
$kalimat = "belajar pemrograman di malas ngoding";  
  
// memisahkan string menjadi array  
$data = explode(" ", $kalimat);  
  
// string menjadi array  
print_r($data);  
//print_r berguna untuk menampilkan isi array  
  
echo "<br/>";  
echo "<br/>";  
  
echo " Data 1 = " . $data[0];  
echo "<br/>";
```

```
echo " Data 2 = " . $data[1];
echo "<br/>";
echo " Data 3 = " . $data[2];
echo "<br/>";
echo " Data 4 = " . $data[3];
echo "<br/>";
echo " Data 5 = " . $data[4];

echo "<br/>";
echo "<br/>";
echo "<h3>Menggabungkan string dengan implode</h3>";
echo "<br/>";

// menggabungkan string kembali dengan fungsi implode()
echo implode(" ",$data);
?>
```

bisa teman-teaman perhatikan pada contoh di atas. variabel data berisi data yang sudah di explode/di pisahkan. untuk menggabungkannya kembali kita bisa menggunakan fungsi implode().

```
// menggabungkan string kembali dengan fungsi implode()
echo implode(" ",$data);
```

## www.malasngoding.com

Array ( [0] => belajar [1] => pemrograman [2] => di [3] => malas [4] => ngoding )

Data 1 = belajar  
Data 2 = pemrograman  
Data 3 = di  
Data 4 = malas  
Data 5 = ngoding

### Menggabungkan string dengan implode

belajar pemrograman di malas ngoding



explode dan implode php

Mungkin sekian dulu penjelasan tentang tutorial Mengenal Fungsi Explode Dan Implode Pada PHP. kita telah membahas tentang pengertian explode dan implode, serta kegunaan dan cara penulisan fungsi explode dan imlode di php.

## **BELAJAR PHP TINGKAT MENENGAH**

# Membuat Kalkulator Sederhana Dengan PHP

Tutorial Membuat Kalkulator Sederhana Dengan PHP ini merupakan tutorial lanjutan dari tutorial sebelumnya di tutorial kategori PHP tentang pengenalan operasi aritmatika atau matematika pada PHP. setelah mengenal macam-macam operasi aritmatika pada PHP maka pada tutorial Membuat Kalkulator Sederhana Dengan PHP ini kita akan mengimplementasikan penggunaan operator aritmatika pada php menjadi sebuah aplikasi sederhana untuk lebih mengasah lagi pengetahuan tentang dasar-dasar PHP dan mencoba melatih logika kita. karena membuat kalkulator sederhana dengan PHP merupakan sebuah aplikasi yang sangat sederhana untuk melatih logika sederhana.

Pada tutorial Membuat Kalkulator Sederhana Dengan PHP ini kita akan mencoba menggabungkan materi-materi php dasar yang sudah kita pelajari pada tutorial sebelumnya di [www.malasngoding.com](http://www.malasngoding.com). di asumsikan untuk mempelajari dan memahami terlebih dahulu tentang dasar-dasar php contohnya seperti menanganan form (Form handling) dengan PHP seperti penanganan form dengan method \$\_GET dan \$\_POST, kemudian lagi yang terpenting lagi pemahaman dasar tentang penggunaan operator aritmatika pada PHP.

Berikut ini adalah beberapa file yang kita butuhkan untuk membuat aplikasi kalkulator sederhana dengan PHP ini.

- index.php
- style.css

Tahap pertama untuk membuat kalkulator dengan PHP yang harus di lakukan adalah kita design dulu form kalkulator nya. buat sebuah file PHP dengan nama index.php dalam folder project anda. lokasi folder project saya untuk membuat contoh kalkulator sederhana ini ada di <http://localhost/malasngoding/kalkulator/>.

## index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Kalkulator Sederhana Dengan PHP | www.malasngoding.com</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
```

```

<?php
if(isset($_POST['hitung'])){
    $bil1 = $_POST['bil1'];
    $bil2 = $_POST['bil2'];
    $operasi = $_POST['operasi'];
    switch ($operasi) {
        case 'tambah':
            $hasil = $bil1+$bil2;
            break;
        case 'kurang':
            $hasil = $bil1-$bil2;
            break;
        case 'kali':
            $hasil = $bil1*$bil2;
            break;
        case 'bagi':
            $hasil = $bil1/$bil2;
            break;
    }
}
?>
<div class="kalkulator">
    <h2 class="judul">KALKULATOR</h2>
    <a class="brand"
    href="https://www.malasngoding.com">www.malasngoding.com</a>
    <form method="post" action="index.php">
        <input type="text" name="bil1" class="bil" autocomplete="off"
placeholder="Masukkan Bilangan Pertama">
        <input type="text" name="bil2" class="bil" autocomplete="off"
placeholder="Masukkan Bilangan Kedua">
        <select class="opt" name="operasi">
            <option value="tambah">+</option>
            <option value="kurang">-</option>
            <option value="kali">x</option>
            <option value="bagi">/</option>
        </select>
        <input type="submit" name="hitung" value="Hitung" class="tombol">
    </form>
    <?php if(isset($_POST['hitung'])){ ?>
        <input type="text" value="<?php echo $hasil; ?>" class="bil">
    <?php } else{ ?>
        <input type="text" value="0" class="bil">
    <?php } ?>
    </div>
</body>
</html>

```

dan berikan sentuhan cssnya

## style.css

```
body{
    background: #F2F2F2;
    font-family: sans-serif;
}

.kalkulator{
    width: 335px;
    background: #2F495A;
    margin: 100px auto;
    padding: 10px 20px 50px 20px;
    border-radius: 5px;
    box-shadow: 0px 10px 20px 0px #D1D1D1;
}

.bil{
    width: 300px;
    margin: 5px;
    border: none;
    font-size: 16pt;
    border-radius: 5px;
    padding: 10px;
}

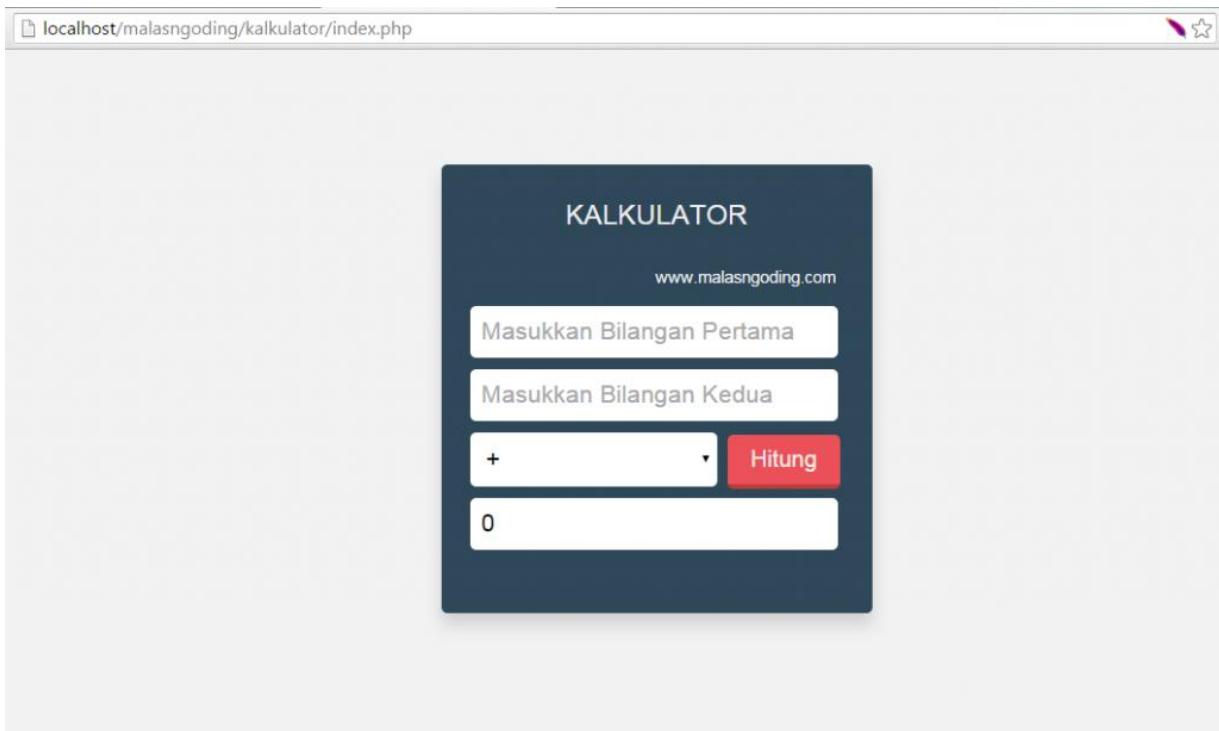
.opt{
    font-size: 16pt;
    border: none;
    width: 215px;
    margin: 5px;
    border-radius: 5px;
    padding: 10px;
}

.tombol{
    background: #EC5159;
    border-top: none;
    border-right: none;
    border-left: none;
    border-radius: 5px;
    padding: 10px 20px;
    color: #eee;
    font-size: 15pt;
    border-bottom: 4px solid #BF3D3D;
}

.brand{
    color: #eee;
    font-size: 11pt;
    float: right;
    text-decoration: none;
    margin: 12px;
}
```

```
.judul{  
    text-align: center;  
    color: #eee;  
    font-weight: normal;  
}
```

dan coba kita jalankan pada browser. <http://localhost/malasngoding/kalkulator/>



membuat kalkulator sederhana dengan php

Akan saya jelaskan sedikit tentang syntax php kalkulator yang saya buat di atas. perhatikan pada bagian form

```
<form method="post" action="index.php">  
    <input type="text" name="bil1" class="bil" autocomplete="off"  
placeholder="Masukkan Bilangan Pertama">  
    <input type="text" name="bil2" class="bil" autocomplete="off"  
placeholder="Masukkan Bilangan Kedua">  
    <select class="opt" name="operasi">  
        <option value="tambah">+</option>  
        <option value="kurang">-</option>  
        <option value="kali">x</option>  
        <option value="bagi">/</option>  
    </select>  
    <input type="submit" name="hitung" value="Hitung" class="tombol">  
</form>
```

saya menggunakan method POST untuk penanganan data bilangan yang di input, pada bilangan satu saya memberikan nama dengan "bil1" dan bilangan dua dengan "bil2". dan bagian operator nya saya memberikan nama operasi yang masing-masing value nya bisa anda perhatikan sendiri. untuk action nya saya membuatnya untuk di proses pada halaman index.php sendiri. bisa juga kalau actionnya di kosongkan maka dengan otomatis action form akan di proses pada file form tersebut sendiri.

kemudian perhatikan pada bagian penangkapan data dari form.

```
<?php
if(isset($_POST['hitung'])) {
    $bil1 = $_POST['bil1'];
    $bil2 = $_POST['bil2'];
    $operasi = $_POST['operasi'];
    switch ($operasi) {
        case 'tambah':
            $hasil = $bil1+$bil2;
            break;
        case 'kurang':
            $hasil = $bil1-$bil2;
            break;
        case 'kali':
            $hasil = $bil1*$bil2;
            break;
        case 'bagi':
            $hasil = $bil1/$bil2;
            break;
    }
}
?>
```

Function isset() berfungsi untuk memeriksa ketersediaan data. jadi di sini saya membuat pengecekan untuk data submit dari form. berarti jika form di submit maka data yang di inputkan dari form saya simpan pada masing-masing variabel.

```
$bil1 = $_POST['bil1'];
$bil2 = $_POST['bil2'];
$operasi = $_POST['operasi'];
```

Selanjutnya saya menggunakan fungsi switch case pada php untuk mengecek operasi yang di pilih saat sebelum form di submit.

```
switch ($operasi) {
    case 'tambah':
        $hasil = $bil1+$bil2;
```

```

        break;
    case 'kurang':
        $hasil = $bil1-$bil2;
        break;
    case 'kali':
        $hasil = $bil1*$bil2;
        break;
    case 'bagi':
        $hasil = $bil1/$bil2;
        break;
}

```

Pengecekan dengan switch case ini saya lakukan untuk memeriksa operator apa yang di pilih untuk menghitung bilangan satu dan bilangan dua yang di input, jadi jika operator yang di pilih adalah tambah berarti bilangan satu di tambah bilangan dua, jika operator yang di pilih adalah kurang maka berarti bilangan satu di kurang bilangan dua begitu juga seterusnya dengan operator kali dan bagi jika di pilih. dan masing-masing operasi yang di lakukan saya rangkum kedalam sebuah variabel hasil dari operasinya seperti yang dapat di lihat pada syntax di atas. saya merangkum hasilnya ke dalam variabel \$hasil.

Sampai di sini berarti kita sudah memiliki hasil dari operasi aritmatikanya. kemudian yang harus kita lakukan adalah menampilkan hasil operasi nya.

```

<?php if(isset($_POST['hitung'])){ ?>
    <input type="text" value="<?php echo $hasil; ?>" class="bil">
<?php }else{ ?>
    <input type="text" value="0" class="bil">
<?php } ?>

```

Kita periksa dulu apakah submit telah di lakukan seperti cara menangkap data dari form di atas tadi. kemudian jika form sudah di submit kan otomatis kita sudah memiliki hasilnya (hasil yang di dapat dari pengecekan menggunakan switch case di atas tadi). tinggal kita tampilkan saja

```
<input type="text" value="<?php echo $hasil; ?>" class="bil">
```

Jika form belum di submit maka yang ingin saya tampilkan adalah nol "0".

```
<input type="text" value="0" class="bil">
```

A screenshot of a web browser window displaying a calculator application. The URL in the address bar is `localhost/malasngoding/kalkulator/index.php`. The page title is "KALKULATOR" and the website URL is `www.malasngoding.com`. There are three input fields: the first contains "15", the second contains "6", and the third contains "x". To the right of the third field is a red button labeled "Hitung". Below these fields is a large empty input field.

Mengisi nilai yang ingin di hitung

A screenshot of a web browser window displaying the same calculator application. The URL in the address bar is `localhost/malasngoding/kalkulator/index.php`. The page title is "KALKULATOR" and the website URL is `www.malasngoding.com`. The input fields now show placeholder text: "Masukkan Bilangan Pertama" and "Masukkan Bilangan Kedua". The third field contains "+" and the red "Hitung" button is visible. Below these is a final input field containing the result "90".

Hasil perhitungan

sekian tutorial membuat kalkulator sederhana dengan PHP.

## Membuat Perhitungan Laba Dan Rugi Dengan PHP

Perhitungan laba dan rugi sangat di butuhkan di dalam membuat aplikasi penjualan. karena aplikasi atau sistem yang kita buat harus mampu mendeteksi laba atau rugi di setiap transaksi penjualan. jadi di setiap penginputan transaksi penjualan oleh kasir, sistem harus mampu mendeteksi apakah yang di dapat laba atau rugi. kalau laba beraa laba nya. dan kalau rugi berapa ruginya.

Jadi di tutorial ini kita hanya membuat simulasi nya saja. maksudnya kita tidak membuat aplikasinya secara komplit. di [www.malasngoding.com](http://www.malasngoding.com) saya sudah share aplikasi penjualan dengan php dan mysql. yang saya buat ketiga saya baru mulai belajar pemrograman php.

Jadi di sini kita hanya membuat logikanya saja bagaimana cara menghitung jumlah laba atau rugi di setiap transaksi dengan php. untuk lebih lanjutnya teman-teman bisa mengembangkan nya lagi sesuai dengan kebutuhan teman-teman. jadi sambil belajar bisa langsung membuat aplikasi penjualan sederhana dengan php dan mysql. lanjut langsung ke tutorial nya . cara mendeteksi laba dan rugi di proses penjualan.

Oke siapkan langsung sebuah file php. di file ini kita akan membuat form sederhana saja. untuk menginput jumlah harga terjual ya dan harga modal nya. setelah di submit maka akan muncul hasil nya berupa laba atau rugi. lengkap dengan jumlah laba atau rugi yang di dapat.

index.php

```
<!DOCTYPE html>
<html>
<head>
<title>Membuat perhitungan laba dan rugi dengan php | www.malasngoding.com</title>
</head>
<body>
<h1>www.malasngoding.com</h1>
<h3>Membuat perhitungan laba rugi dengan PHP</h3>
<br/>
<form method="post" action="hitung.php">
<table>
<tr>
<td>Harga Jual</td>
<td><input type="text" name="harga"></td>
</tr>
<tr>
```

```
<td>Modal</td>
<td><input type="text" name="modal"></td>
</tr>
<tr>
<td></td>
<td><input type="submit" value="Hitung!"></td>
</tr>
</table>
</form>
</body>
</html>
```

## hitung.php

```
<?php
$harga = $_POST['harga'];
$modal = $_POST['modal'];

if($harga > $modal){
    $laba = $harga-$modal;
    echo "Anda Mendapat LABA Sebesar = " . $laba;
}elseif($modal > $harga) {
    $rugi = $modal-$harga;
    echo "Anda Mendapat RUGI Sebesar = " . $rugi;
} else{
    echo "Tidak laba tidak rugi";
}
?>
```

akan saya jelaskan sedikit. terdapat 2 buah form yang sudah kita beri nama dengan harga dan laba.

```
<input type="text" name="harga">
<input type="text" name="modal">
```

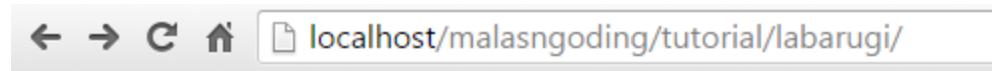
lalu pada saat form di submit, kita tangkap data ny. agar lebih mudah kita proses di kemudiannya.

```
$harga = $_POST['harga'];
$modal = $_POST['modal'];
```

Lalu kita cek laba dan ruginya. jika harga lebih besar dari modal berarti kita mendapat laba. dan jika modal lebih besar dari harga berarti kita rugi. untuk mendapatkan jumlah laba nya berarti harga di kurang dengan modal. dan untuk mendapatkan jumlah ruginya berarti modal di kurang harga.

```
if($harga > $modal){  
    $laba = $harga-$modal;  
    echo "Anda Mendapat LABA Sebesar = " . $laba;  
}elseif($modal > $harga){  
    $rugi = $modal-$harga;  
    echo "Anda Mendapat RUGI Sebesar = " . $rugi;  
}else{  
    echo "Tidak laba tidak rugi";  
}
```

sekarang coba kita isikan misalnya harga yang terjualnya sebesar 3000. dan modal nya 1000. maka otomatis kan kita mendapat laba sebesar 2000.



## www.malasngoding.com

### Membuat perhitungan laba rugi dengan PHP

Harga Jual	<input type="text" value="3000"/>
Modal	<input type="text" value="1000"/>
<input type="button" value="Hitung!"/>	

membuat perhitungan laba dan rugi dengan php

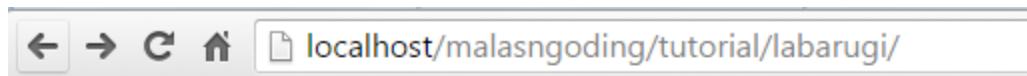
Hasilnya.



Anda Mendapat LABA Sebesar = 2000

membuat perhitungan laba dengan php

dan bagaimana jika kita mengisi harga terjualnya misalnya 5000. sedangkan modalnya 7000. maka kita akan rugi sebesar 2000.



## **www.malasngoding.com**

### **Membuat perhitungan laba rugi dengan PHP**

Harga Jual	<input type="text" value="5000"/>
Modal	<input type="text" value="7000"/>
<input type="button" value="Hitung!"/>	

membuat perhitungan rugi dengan php

hasilnya.



Anda Mendapat RUGI Sebesar = 2000

perhitungan rugi dengan php

Selesai. sekian dulu tutorial Membuat Perhitungan Laba Dan Rugi Dengan PHP. semoga dapat bermanfaat.

# Membuat Format Tanggal Indonesia Dengan PHP

Pada tutorial kali ini, kita akan belajar bagaimana cara membuat format tanggal indonesia dengan PHP (date format indonesia).

Pada dasarnya, penulisan tanggal pada php atau mysql adalah dengan format YYYY-MM-DD atau tahun-bulan-tanggal. contohnya 2017-10-21. sedangkan kita di indonesia menggunakan format tanggal dengan DD-MM-YYYY atau misalnya 21-10-2017 atau 21 Oktober 2017. nah jadi bagaimana sih cara membuatnya ? caranya mudah. yuk simak cara membuat format tanggal indonesia dengan PHP berikut ini.

## Mengubah Format Tanggal Dengan PHP

Untuk mengubah bentuk format tanggal dengan php, kita bisa menggunakan syntax berikut.

```
<h2>Format Tanggal PHP | www.malasngoding.com</h2>

<?php
// Tanggal hari ini
echo date('d F Y'); // Hasilnya tanggal hari ini (21 October 2017)

echo "<br/>";

echo date('d F Y', strtotime('1994-02-15')); // Hasil 15 February 1994
```

dan hasilnya



## Format Tanggal PHP | www.malasngoding.com

21 October 2017  
15 February 1994



Coba teman-teman perhatikan pada contoh di atas. untuk mengubah format tanggal kita bisa menggunakan strtotime pada parameter kedua di function date(). strtotime() adalah fungsi untuk mengubah string ke waktu(time). jadi tanggal yang ditentukan bisa di letakkan pada parameter kedua seperti contoh di atas.

```
echo date('d F Y', strtotime('1994-02-15')); // Hasil 15 February 1994
```

Oke, Sampai di sini kita telah berhasil mengubah format tanggal dengan PHP. tapi masih dalam bentuk format bahasa inggris. sekarang kita akan membuat format tanggal ke dalam bahasa indonesia.

Untuk membuat format tanggal indonesia dengan PHP, kita akan menggunakan function php. perhatikan syntax berikut ini dan penjelasannya.

### tgl\_indo.php

```
<h2>Format Tanggal PHP | www.malasngoding.com</h2>
<?php
function tgl_indo($tanggal){
$bulan = array (
1 => 'Januari',
'Februari',
'Maret',
'April',
'Mei',
'Juni',
'Juli',
'Agustus',
'September',
'Oktober',
'November',
'Desember'
);
$pecahkan = explode('-', $tanggal);
// variabel pecahkan 0 = tanggal
// variabel pecahkan 1 = bulan
// variabel pecahkan 2 = tahun

return $pecahkan[2] . ' ' . $bulan[ (int)$pecahkan[1] ] . ' ' . $pecahkan[0];
}

echo tgl_indo(date('Y-m-d')); // 21 Oktober 2017
echo "<br/>";
echo "<br/>";

echo tgl_indo("1994-02-15"); // 15 Februari 1994
```

maka hasilnya

← → ⌂ ⓘ localhost/tgl\_indo.php

## Format Tanggal PHP | www.malasngoding.com

21 Oktober 2017

15 Februari 1994

format tanggal indonesia

Coba teman-teaman perhatikan inti dari fungsi tgl\_indo() yang kita buat di atas.

```
function tgl_indo($tanggal) {
    $bulan = array (
        1 => 'Januari',
        'Februari',
        'Maret',
        'April',
        'Mei',
        'Juni',
        'Juli',
        'Agustus',
        'September',
        'Oktober',
        'November',
        'Desember'
    );

    $pecahkan = explode('-', $tanggal);
    // variabel pecahkan 0 = tahun
    // variabel pecahkan 1 = bulan
    // variabel pecahkan 2 = tanggal

    return $pecahkan[2] . ' ' . $bulan[ (int)$pecahkan[1] ] . ' ' .
$pecahkan[0];
}
```

Data tanggal yang kita tangkap, kita pecahkan menjadi array dengan menggunakan fungsi explode(). sehingga data tanggal akan berada di \$pecahkan[2], data bulan berada di \$pecahkan[1] dan data tahun berada pada \$pecahkan[0].

Kemudian baru kita balik urutannya. dan untuk membuat nama bulan menjadi bahasa indonesia,kita panggil variabel array bulan sesuai urutan.

# Membuat Format Hari Pada PHP

Pasti teman-teman semua sering menggunakan function date('D') untuk menampilkan nama hari dengan php kan? tapi nama yang muncul otomatis pasti dalam bahasa inggris.

Kemarin ada salah satu dari teman-teman yang sering membaca tutorial PHP di malasngoding.com, beliau bertanya bagaimana cara menampilkan nama hari dengan php. tapi dalam bahasa indonesia. kebetulan sekali, sekalian saja saya buatkan tutorial nya agar teman-teman yang lain juga bisa membaca dan mempelajarinya. sehingga ladang amal semakin luas.

Langsung saja kita mulai tentang cara membuat format hari indonesia dengan PHP.

Untuk mengikuti tutorial format hari indonesia php ini, silahkan teman-teman buat sebuah file php baru. di sini saya membuat file php baru dengan nama hari.php. lalu buat sebuah function dengan nama hari\_ini(). nanti akan di jelaskan lebih mendetail pada bagian bawah tutorial ini.

hari.php

```
<?php

// function untuk menampilkan nama hari ini dalam bahasa indonesia
// di buat oleh malasngoding.com

function hari_ini(){
$hari = date ("D");

switch($hari){
case 'Sun':
$hari_ini = "Minggu";
break;

case 'Mon':
$hari_ini = "Senin";
break;

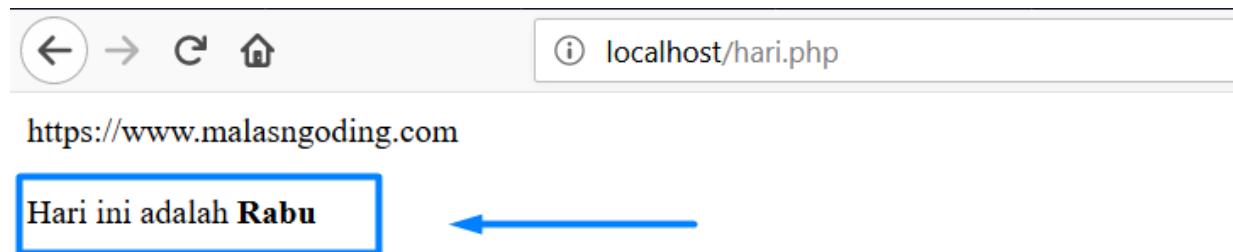
case 'Tue':
$hari_ini = "Selasa";
break;

case 'Wed':
$hari_ini = "Rabu";
break;
}
```

```
case 'Thu':  
$hari_ini = "Kamis";  
break;  
  
case 'Fri':  
$hari_ini = "Jumat";  
break;  
  
case 'Sat':  
$hari_ini = "Sabtu";  
break;  
default:  
$hari_ini = "Tidak di ketahui";  
break;  
}  
  
return "<b>" . $hari_ini . "</b>";  
}  
  
echo "Hari ini adalah ". hari_ini();  
?>
```

sekarang coba teman-teman jalankan hasilnya pada browser. karena file hari.php ini saya simpan di directory root localhost saya, maka untuk mengaksesnya, saya mengaksesnya melalui browser dengan alamat 'localhost/hari.php'.

Maka hasilnya akan tampil nama hari ini yaitu hari rabu. karena hari ini adalah tanggal 13-agustus-2017.



format tanggal indonesia php

nama hari pun sukses kita tampilkan dalam bahasa indonesia.

## Penjelasan Membuat Format Hari Indonesia Dengan PHP

Jika teman-teman perhatikan pada source code hari.php yang saya bagikan di atas, teman-teman akan melihat bahwa tahap pertama yang kita lakukan adalah membuat sebuah function baru.

Di sini kita membuat function nya dengan nama hari\_ini(). yang mana pada function ini kita membuat menyimpan data hari ini ke dalam variabel \$hari.

kemudian kita menyeleksi isi dari variabel \$hari dengan menggunakan switch case.

```
function hari_ini(){
    $hari = date ("D");

    switch($hari){
        case 'Sun':
            $hari_ini = "Minggu";
            break;

        case 'Mon':
```

```
$hari_ini = "Senin";
break;

case 'Tue':
$hari_ini = "Selasa";
break;

case 'Wed':
$hari_ini = "Rabu";
break;

case 'Thu':
$hari_ini = "Kamis";
break;

case 'Fri':
$hari_ini = "Jumat";
break;

case 'Sat':
$hari_ini = "Sabtu";
break;
default:
$hari_ini = "Tidak di ketahui";
break;
}

return "<b>" . $hari_ini . "</b>";
}
```

terakhir, kita tinggal mengembalikan nama hari yang berhasil di dapatkan.

```
return "<b>" . $hari_ini . "</b>";
```

dan selesai. kita tinggal memanggil function hari\_ini() untuk menampilkan nama hari dalam bahasa indonesia.

Sekian tutorial Membuat Format Hari Indonesia Pada PHP.

## Membuat Format Rupiah Di PHP

Setelah sebelumnya kita belajar membuat format hari dengan PHP. pada tutorial kali ini kita akan belajar membuat Membuat Format Rupiah Di PHP. Cara termudah membuat format rupiah di php adalah dengan cara membuat sebuah function yang akan menangani atau mengubah dari angka menjadi nominal angka dengan format rupiah. penasaran dengan caranya? yuk simak penjelasan berikut.

Langkah awal, yuk kita buat sebuah file php sebagai contoh. di sini saya membuat sebuah file php dengan nama rupiah.php. dan ketikkan syntax berikut ini. ingat ya, saya sarankan kepada teman-teman untuk mengetik sendiri syntax nya. karena dengan sering-sering mengetik syntax, kita akan terbiasa dan mudah mengingat. dari pada di copas.

Lebih detailnya akan dijelaskan pada bagian akhir tutorial ini.

rupiah.php

```
<?php
//membuat format rupiah dengan PHP
//tutorial www.malasngoding.com

function rupiah($angka){
    $hasil_rupiah = "Rp " . number_format($angka,2,',','.');
    return $hasil_rupiah;

}

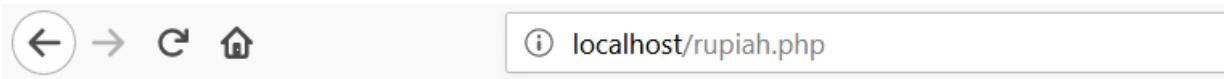
echo rupiah(1000000);
?>
```

Sekarang coba jalankan pada browser. karena file php yang saya buat bernama rupiah.php dan mengimpannya di direktori root localhost (folder htdocs), maka untuk menjalankannya saya mengakses alamat " localhost/rupiah.php ".

Pada syntax di atas kita menetapkan nilai yang akan di ubah ke rupiah adalah "1000000".

```
echo rupiah(1000000);
```

localhost/rupiah.php



www.malasngoding.com

Rp 1.000.000,00

membuat format rupiah di php

seperti yang terlihat di atas. hasilnya akan menjadi " Rp. 1.000.000,00 ".

### Penjelasan Membuat Format Rupiah Di PHP

coba teman-teaman perhatikan.

```
function rupiah($angka){  
    $hasil_rupiah = "Rp " . number_format($angka,2,',','.') ;  
    return $hasil_rupiah;  
}
```

Pertama kita membuat sebuah function dengan nama rupiah(). dan isinya adalah, kita menangkap data yang dikirim ke function ini dan memasukkannya dalam variabel \$angka. selanjutnya kita gunakan function number\_format() yang telah di jelaskan pada tutorial sebelumnya. untuk membuat angka menjadi berformat angka (menggunakan titik untuk menentukan angka puluhan, ratusan dan ribuan).

Kita juga menambahkan "Rp." di awalnya. dan terakhir kita tinggal me return hasilnya untuk di tampilkan.

dan selesai.

Teman-teman juga bisa mencoba memasukkan angka dengan nominal yang lain. misalnya " 1528700 ".

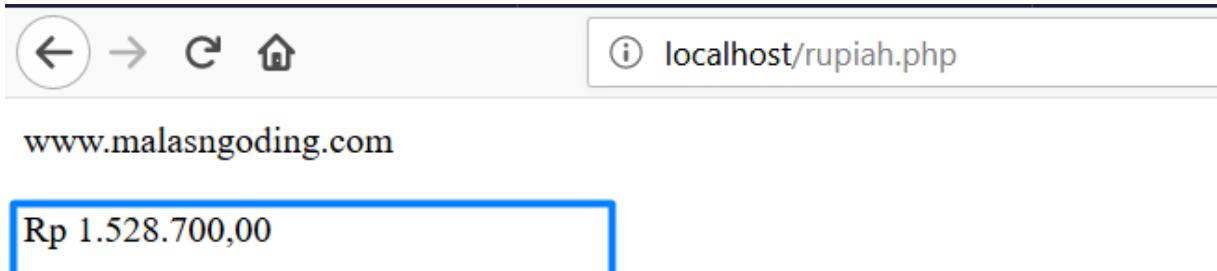
```
<?php
//membuat format rupiah dengan PHP
//tutorial www.malasngoding.com

function rupiah($angka){
    $hasil_rupiah = "Rp " . number_format($angka,2,',','.');
    return $hasil_rupiah;

}

echo rupiah(1528700);
?>
```

maka hasilnya :



format rupiah php

Selesai.

# Membuat Fungsi Terbilang Dengan PHP

Pada artikel ini kita akan sedikit membahas tentang cara membuat fungsi terbilang dengan PHP secara mudah. Pasti teman-teman penasaran dengan caranya, yuk kita simak bagaimana sih cara membuat fungsi terbilang di PHP, dan untuk apa sih gunanya kalau mungkin ada teman-teman yang belum tau.

Pertama kita bahas dulu yang dimaksud dengan terbilang itu apa? kalau di perhitungan jumlah nominal angka, maka terbilang itu adalah Penyebutan nominal angka. misalnya kita memiliki angka 5.341, maka terbilang “Lima ribu tiga ratus empat puluh satu”.

Oke? pengertiannya udah, sekarang yang menjadi permasalahan bagi kita sebagai programmer adalah bagaimana cara membuat fungsi terbilang seperti di atas? jika mungkin kita sedang berhadapan dengan tugas kuliah, atau pun project yang mengharuskan kita untuk membuat fungsi terbilang dengan PHP.

Misalnya ada pengelolaan data keuangan, yang nominal uangnya harus dibuatkan fitur terbilang dari nominal keuangannya. yuk simak cara mudah membuat fungsi terbilang dengan PHP.

Untuk mengikuti tutorial ini, Seperti biasa, silahkan teman-teman buat sebuah file php. di sini saya membuat sebuah file PHP dengan nama *terbilang.php*. nanti teman-teman bisa menerapkan function ini ke project teman-teman atau membuat library sendiri.

**terbilang.php**

```
<!DOCTYPE html>
<html>
<head>
    <title>Cara mudah membuat fungsi terbilang dengan PHP | www.malasngoding.com</title>
</head>
<body>

    <h1>Membuat Fungsi Terbilang Dengan PHP | WWW.MALASNGODING.COM</h1>

    <?php

    // FUNGSI TERBILANG OLEH : MALASNGODING.COM
    // WEBSITE : WWW.MALASNGODING.COM
    // AUTHOR : https://www.malasngoding.com/author/admin
```

```

function penyebut($nilai) {
    $nilai = abs($nilai);
    $huruf = array("", "satu", "dua", "tiga", "empat", "lima", "enam",
    "tujuh", "delapan", "sembilan", "sepuluh", "sebelas");
    $temp = "";
    if ($nilai < 12) {
        $temp = " ". $huruf[$nilai];
    } else if ($nilai <20) {
        $temp = penyebut($nilai - 10). " belas";
    } else if ($nilai < 100) {
        $temp = penyebut($nilai/10)." puluh". penyebut($nilai % 10);
    } else if ($nilai < 200) {
        $temp = " seratus" . penyebut($nilai - 100);
    } else if ($nilai < 1000) {
        $temp = penyebut($nilai/100) . " ratus" . penyebut($nilai % 100);
    } else if ($nilai < 2000) {
        $temp = " seribu" . penyebut($nilai - 1000);
    } else if ($nilai < 1000000) {
        $temp = penyebut($nilai/1000) . " ribu" . penyebut($nilai % 1000);
    } else if ($nilai < 1000000000) {
        $temp = penyebut($nilai/1000000000) . " milyar" .
penyebut(fmod($nilai,1000000000));
    } else if ($nilai < 1000000000000) {
        $temp = penyebut($nilai/1000000000000) . " trilyun" .
penyebut(fmod($nilai,1000000000000));
    }
    return $temp;
}

function terbilang($nilai) {
    if($nilai<0) {
        $hasil = "minus ". trim(penyebut($nilai));
    } else {
        $hasil = trim(penyebut($nilai));
    }
    return $hasil;
}

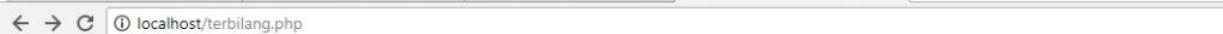
$angka = 1530093;
echo terbilang($angka);
?>
</body>
</html>

```

kemudian simpan dan jalankan file terbilang.php tersebut.

di bawah akan saya jelaskan cara membuat terbilang dengan php ini secara bertahap.

dan hasilnya



## Membuat Fungsi Terbilang Dengan PHP | WWW.MALASNGODING.COM

satu juta lima ratus tiga puluh ribu sembilan puluh tiga

cara mudah membuat fungsi terbilang dengan php

Dan hasilnya pun muncul.

### Penjelasan

Coba teman-teman perhatikan pada file terbilang.php tadi. pada variabel angka kita memiliki nilai 1530093. dan hasilnya menjadi "*satu juta lima ratus tiga puluh ribu sembilan puluh tiga*".

```
$angka = 1530093;
```

Selanjutnya variabel \$angka yang sudah di isi dengan nilai tadi kita panggil dengan menggunakan function terbilang() yang sudah kita buat.

```
echo terbilang($angka);
```

nah, di function terbilang() tadi kita cek dulu apakah bilangannya merupakan bilangan positif atau negatif(minus).

```
function terbilang($nilai) {
    if($nilai<0) {
        $hasil = "minus ". trim(penyebut($nilai));
    } else {
        $hasil = trim(penyebut($nilai));
    }
    return $hasil;
}
```

Kemudian dari dalam function terbilang() ini kita panggil lagi function penyebut(). untuk memisahkan nilai nya apakah menjadi satuan, puluhan, ratusan, ribuan, ratus ribuan dan seterusnya. dan terakhir tinggal me-return hasilnya.

```
function penyebut($nilai) {
    $nilai = abs($nilai);
    $huruf = array("", "satu", "dua", "tiga", "empat", "lima", "enam",
    "tujuh", "delapan", "sembilan", "sepuluh", "sebelas");

    $temp = "";
    if ($nilai < 12) {
        $temp = " ". $huruf[$nilai];
    } else if ($nilai < 20) {
        $temp = penyebut($nilai - 10). " belas";
    } else if ($nilai < 100) {
        $temp = penyebut($nilai/10). " puluh". penyebut($nilai % 10);
    } else if ($nilai < 200) {
        $temp = " seratus" . penyebut($nilai - 100);
    } else if ($nilai < 1000) {
        $temp = penyebut($nilai/100) . " ratus" . penyebut($nilai % 100);
    } else if ($nilai < 2000) {
        $temp = " seribu" . penyebut($nilai - 1000);
    } else if ($nilai < 1000000) {
        $temp = penyebut($nilai/1000) . " ribu" . penyebut($nilai % 1000);
    } else if ($nilai < 1000000000) {
        $temp = penyebut($nilai/1000000000) . " milyar" .
penyebut(fmod($nilai,1000000000));
    } else if ($nilai < 1000000000000000) {
        $temp = penyebut($nilai/1000000000000000) . " trilyun" .
penyebut(fmod($nilai,1000000000000000));
    }
    return $temp;
}
```

## Penjelasan Function PHP

Berikut ini saya sertakan penjelasan tentang function-function php yang kita gunakan pada tutorial ini.

Nama Function	Pengertian
abs()	Function ini di gunakan untuk mengembalikan nilai positif. misal yg di input di parameternya adalah bilangan minus, maka akan di kembalikan menjadi bilangan positif.
fmod()	Function ini di gunakan untuk membuat perhitungan modulus.

trim()	Function untuk menghilangkan character yg di inginkan.
--------	--

## Membuat Halaman Web Dinamis Dengan PHP

Pada artikel tutorial PHP ini kita akan membahas tentang cara membuat halaman web dinamis dengan php. atau bisa juga kita sebuah membuat halaman template web yang bersifat dinamis dengan php.

Apa yang di maksud dengan halaman web dinamis? dan apa sih kegunaan dan keuntungan membuat halaman web yang dinamis? yuk simak penjelasan tentang tutorial cara membuat halaman dinamis dengan php berikut ini.

### Kegunaan Halaman Web Dinamis ?

Yuk kita bahas sedikit tentang apa sih kegunaannya halaman web dinamis? karena mungkin ada teman-teman yang belum tau.

Halaman web yang bersifat dinamis dapat sangat berguna dan menghemat segala keperluan. contohnya untuk mengedit suatu halaman, kita hanya perlu mengeditnya di halaman itu saja. tanpa harus mengubah di semua halaman.

## Membuat Halaman Web Dinamis Dengan PHP

Berikut ini adalah cara membuat halaman dinamis dengan PHP. Silahkan teman-teman ikuti, dan akan di jelaskan secara detail.

buat sebuah project di localhost. di sini saya membuat sebuah project (folder) dengan nama “dinamis\_php”.

C:/xampp/htdocs/dinamis\_php

Kemudian buat sebuah file dengan nama index.php

index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Halaman Web Dinamis Dengan PHP | www.malasngoding.com</title>
    <!-- menghubungkan dengan file css -->
    <link rel="stylesheet" type="text/css" href="style.css">
    <!-- menghubungkan dengan file jquery -->
    <script type="text/javascript" src="jquery.js"></script>
</head>
<body>
<!--
Author : diki alfarabi hadi
Site : www.malasngoding.com
-->
<div class="content">
    <header>
        <h1 class="judul">WWW.MALASNGODING.COM</h1>
        <h3 class="deskripsi">Membuat Halaman Web Dinamis Dengan PHP</h3>
    </header>

    <div class="menu">
        <ul>
            <li><a href="index.php?page=home">HOME</a></li>
            <li><a href="index.php?page=tentang">TENTANG</a></li>
            <li><a href="index.php?page=tutorial">TUTORIAL</a></li>
        </ul>
    </div>

    <div class="badan">

        <?php
        if(isset($_GET['page'])){
            $page = $_GET['page'];

            switch ($page) {
                case 'home':
                    include "halaman/home.php";
                    break;
                case 'tentang':
                    include "halaman/tentang.php";
                    break;
                case 'tutorial':
                    include "halaman/tutorial.php";
                    break;
                default:
                    echo "<center><h3>Maaf. Halaman tidak di temukan !</h3></center>";
            }
        }
        ?>
    </div>
</div>

```

```

                break;
            }
        }else{
            include "halaman/home.php";
        }

    ?>

    </div>
</div>
</body>
</html>

```

kemudian buat file css nya. agar tampilan nya lebih menarik di pandang.

### style.css

```

body{
    background-color:#f9f9f9;
    font-size:16px
    color:#444;
    font-family: sans-serif;
}

.content{
    width: 80%;
    margin: 10px auto;
}

/*header*/
header{
    background-color: white;
    padding: 20px 10px;
    border-radius: 5px;
    border: 1px solid #f0f0f0;
    margin-bottom: 10px;
}

header h1.judul,
header h3.deskripsi{
    text-align: center;
}

/*menu navigasi*/
.menu{
    background-color: #87CEFA;
    border: 1px solid #f0f0f0;
    border-radius: 8px;
    margin-bottom: 10px;
}

div.menu ul {
    list-style:none;
    overflow: hidden;
}

```

```

}

div.menu ul li {
    float:left;
    text-transform:uppercase;
}

div.menu ul li a {
    display:block;
    padding:0 20px;
    text-decoration:none;
    color:#2c2c2c;
    font-family: sans-serif;
    font-size:13px;
    font-weight:400;
    transition:all 0.3s ease-in-out;
}

div.menu ul li a:hover,
div.menu ul li a.hoverover {
    cursor: pointer;
    color:#fff;
}

div.badan{
    background-color: white;
    border-radius: 5px;
    border: 1px solid #f0f0f0;
    margin-bottom: 10px;
}

div.halaman{
    text-align: center;
    padding: 30px 20px;
}

```

Sampai di sini kita telah selesai membuat kerangka dari halaman web dinamis yang ingin kita buat. selanjutnya kita tinggal membuat sebuah folder bernama "halaman".

di dalam folder ini lah akan kita letakkan halaman-halaman apa saja yang ingin kita buat.

berikut adalah halaman-halaman yang akan kita buat:

- home.php
- tentang.php
- tutorial.php

di contoh tutorial ini saya membuat halaman dengan nama home.php.

### home.php

```
<div class="halaman">
    <h2>Halaman Depan</h2>
    <p>Selamat datang di www.malasngoding.com. ini merupakan halaman depan pada demo cara Membuat Halaman Web Dinamis Dengan PHP</p>
</div>
```

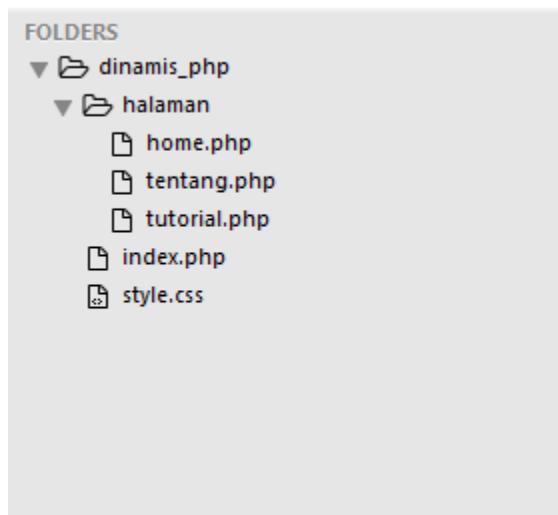
### tentang.php

```
<div class="halaman">
    <h2>Tentang Kami</h2>
    <p>Ini adalah halaman tentang kami</p>
    <p>Malasngoding.com adalah situs yang menyediakan konten-konten berkualitas untuk belajar pemrograman web</p>
</div>
```

### tutorial.php

```
<div class="halaman">
    <h2>Halaman Tutorial</h2>
    <p>Ini adalah halaman tutorial</p>
</div>
```

oke selesai. berikut ini struktur file nya setelah selesai.



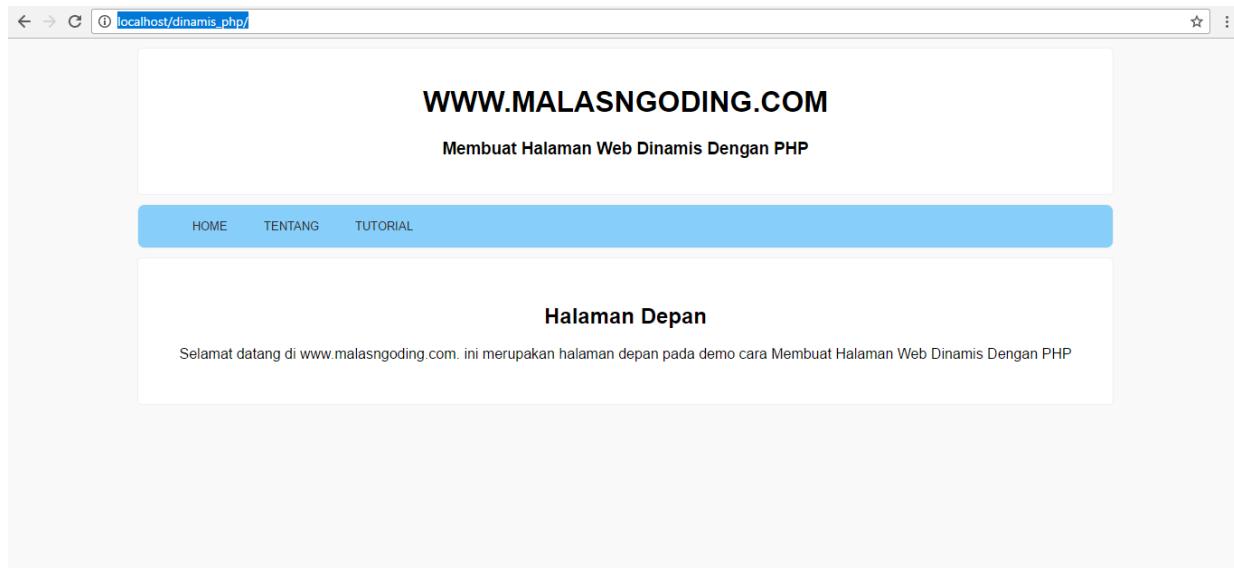
membuat halaman web dinamis dengan php

nah. ayo kita lihat hasil dari halaman dinamis yang telah kita buat dengan php.

sekarang akses project *dinamis\_php* yang telah kita buat di browser.

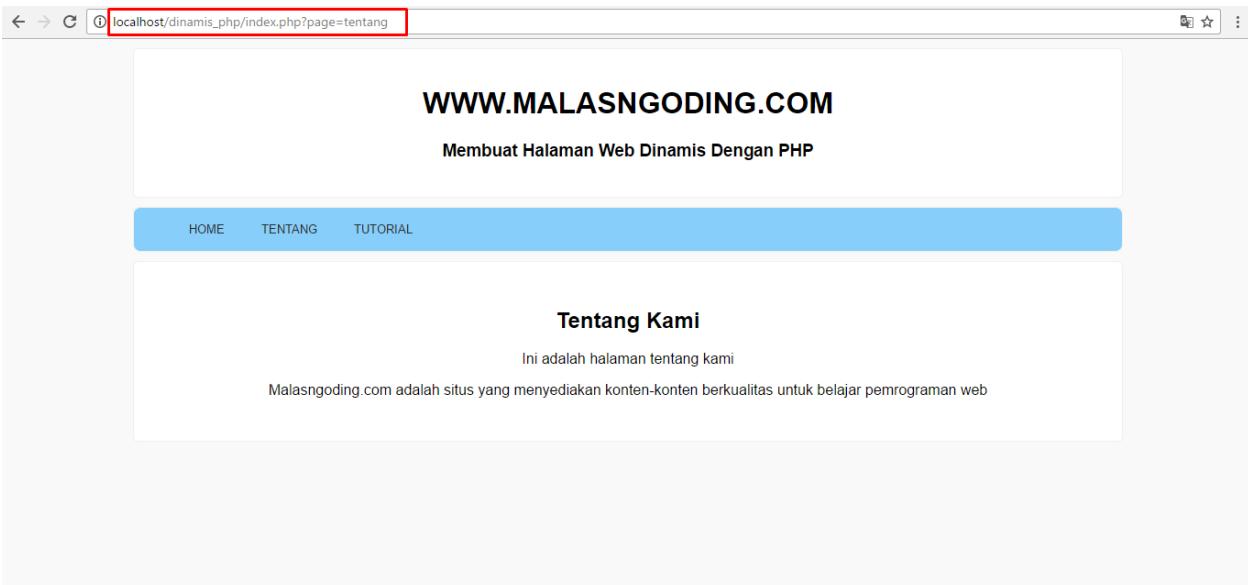
*localhost/dinamis\_php*

dan hasilnya



cara membuat halaman dinamis dengan php

kemudian coba klik pada menu tentang atau tutorial.

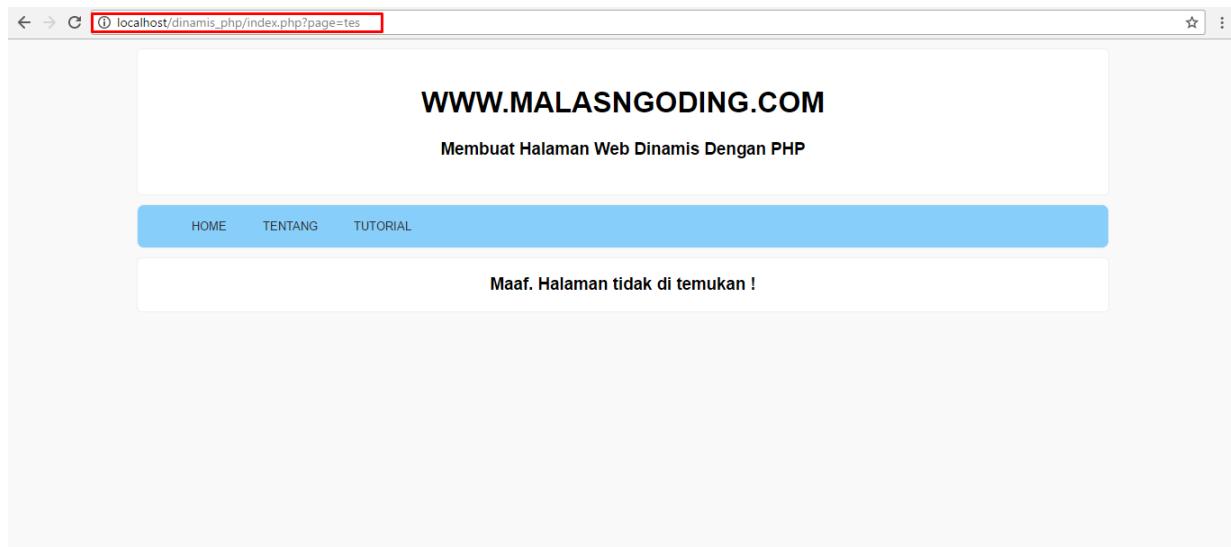


halaman web dinamis php

Kemudian coba juga mengubah halaman url nya menjadi

*localhost/dinamis\_php/index.php?page=tes*

maka akan muncul pemberitahuan bahwa halaman tidak di temukan.



membuat halaman tidak di temukan dengan php

jadi di sini kita sekalian membuat halaman tidak di temukan dengan php. nah selesai sudah halaman dinamis yang telah kita buat dengan php. Jika teman-teman masih bingung dengan cara kerjanya. yuk simak penjelasan berikut.

## Penjelasan

Sebenarnya konsep nya mudah saja teman-teman. coba teman-teman perhatikan pada link url menu yang kita buat di atas.

```
<li><a href="index.php?page=home">HOME</a></li>
<li><a href="index.php?page=tentang">TENTANG</a></li>
<li><a href="index.php?page=tutorial">TUTORIAL</a></li>
```

Nah di sini kita mengalihkan halaman nya dengan menggunakan bantuan dari method GET. jika teman-teman belum paham tentang penggunaan method GET, teman-teman bisa membaca tutorial kita sebelumnya tentang Method GET pada PHP.

di sini kita mengirim data GET dengan nama 'page'.

kemudian kita menangkap data page yang di kirim. dan memeriksa isi dari GET page yang di kirim. dan meng include halaman yang di tuju.

```

<?php
if(isset($_GET['page'])) {
    $page = $_GET['page'];

    switch ($page) {
        case 'home':
            include "halaman/home.php";
            break;
        case 'tentang':
            include "halaman/tentang.php";
            break;
        case 'tutorial':
            include "halaman/tutorial.php";
            break;
        default:
            echo "<center><h3>Maaf. Halaman tidak di temukan !</h3></center>";
            break;
    }
} else{
    include "halaman/home.php";
}

?>

```

di sini kita cek dulu ada atau tidak nya GET page.

```
if(isset($_GET['page'])) {
```

jika tidak ada maka halaman yang di tampilkan adalah halaman home.

```
include "halaman/home.php";
```

jika ada data GET yang di kirim, kemudian kita cek lagi dengan bantuan switch case php.

```

switch ($page) {
    case 'home':
        include "halaman/home.php";
        break;
    case 'tentang':
        include "halaman/tentang.php";
        break;
    case 'tutorial':
        include "halaman/tutorial.php";
        break;
    default:
        echo "<center><h3>Maaf. Halaman tidak di temukan !</h3></center>";
        break;
}

```

jika halaman yang di cari tidak ada atau tidak sesuai maka kita tampilkan pemberitahuan bahwa halaman tidak di temukan.

```
echo "<center><h3>Maaf. Halaman tidak di temukan !</h3></center>";
```

Nah, Selesai.

# Membuat Form Validasi dengan PHP

Form validasi atau form validation, sangat di perlukan di dalam sebuah aplikasi. karena terkadang kita harus membuat sebuah form yang datanya wajib di isi. bisa itu semua form yang wajib di isi atau beberapa form saja wajib atau harus yang di isi. jika form tidak di isi maka akan muncul pesan peringatan bahwa form tersebut belum di isi dan harus di isi. itulah yang akan kita bahas di tutorial kali ini. silahkan teman-teman perhatikan penjelasan berikut ini tentang cara membuat form validasi dengan PHP.

Sebenarnya mudah saja membuat form validasi dengan PHP. teman-teman harus sudah paham dulu tentang penggunaan `$_GET` dan `$_POST` di dalam pengolahan form. atau form handling. silahkan teman-teman baca tutorial sebelumnya tentang pengolahan/penanganan form (Form Handling) dengan php.

Langsung saja mulai tentang cara membuat form validasi di php. pertama di contoh ini saya membuat sebuah file php dengan nama index.php. di file index.php ini saya membuat sebuah form yang mengharuskan untuk mengisi form tersebut.

index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Form Validasi Dengan PHP | www.malasngoding.com</title>
</head>
<body>
    <h1>Membuat Form Validasi Dengan PHP <br/> www.malasngoding.com</h1>

<?php
    if(isset($_GET['nama'])){
        if($_GET['nama'] == "kosong"){
            echo "<h4 style='color:red'>Nama Belum Di Masukkan !</h4>";
        }
    }
?>

<h4>Masukkan Nama Anda :</h4>
<form action="cek.php" method="post">
    <table>
        <tr>
            <td>Nama</td>
            <td><input type="text" name="nama"></td>
            <td><input type="submit" value="Cek"></td>
        </tr>
    </table>
</form>
```

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



# Membuat Form Validasi Dengan PHP

## www.malasngoding.com

Masukkan Nama Anda :

Nama  Cek

cara membuat form validasi dengan php

Coba teman-teaman perhatikan pada form validation index.php. kita membuat pemrosesan form nya di cek.php, dengan menggunakan method post.

```
<form action="cek.php" method="post">
```

Jadi sekarang kita akan membuat file cek.php untuk kita jadikan sebagai pemroses form validation yang kita buat. jadi nantinya di file cek.php ini akan kita buat pengecekan apakah form di isi atau tidak. jika form tidak di isi maka akan di munculkan pesan peringatan.

cek.php

```
<?php  
$nama = $ POST['nama'];  
  
if($nama == "") {
```

```
header("location:index.php?nama=kosong");
}else{
    echo "Nama anda adalah". $nama;
}
?>
```

Seperti yang teman-teman lihat. pertama kita tangkap dulu data yang di kirim dari form. yaitu data nama. kemudian saya memasukkannya ke variabel. agar lebih mudah kita gunakan.

Lalu kita cek apakah data yang di kirimkan dari form tersebut berisi atau tidak. jika form memiliki isi maka kita akan menampilkan nama anda adalah (nama yang di input). dan jika nama tidak di isi maka kita aligkan halaman kembali ke form (index.php). dengan membuat parameter nama=kosong. ini berguna untuk kita tampilkan pesan peringatan bahwa form tersebut harus di isi.

Sekarang kita akan mencoba menjalankannya. kita langsung klik cek tanpa mengisi form nama. maka akan di tampilkan pesan bahwa form nama harus di isi atau belum di isi.



## Membuat Form Validasi Dengan PHP

### www.malasngoding.com

**Nama Belum Di Masukkan !**

**Masukkan Nama Anda :**

Nama  Cek

menampilkan pesan error jika form tidak di isi

Nah pesan bahwa form nama belum di isi atau di masukkan pun telah tampil. dari mana asal pesan ini? yap betul sekali. perhatikan di index.php

```
<?php  
if(isset($_GET['nama'])){  
    if($_GET['nama'] == "kosong"){  
        echo "<h4 style='color:red'>Nama Belum Di Masukkan !</h4>";  
    }  
}  
?>
```

kita cek dulu apakah ada parameter yang di kirim melalui url. dan jika parameter nama sama dengan kosong maka akan di tampilkan pesan nama belum di masukkan.

```
if($_GET['nama'] == "kosong"){  
    echo "<h4 style='color:red'>Nama Belum Di Masukkan !</h4>";  
}
```



**Nama Belum Di Masukkan !**

**Masukkan Nama Anda :**

Nama  Cek

cara membuat form validation dengan php

dan coba kita tes jika kita memasukkan nama dan klik cek.



# Membuat Form Validasi Dengan PHP

## www.malasngoding.com

Masukkan Nama Anda :

Nama

membuat form validation dengan php



Nama anda adalah Malas Ngoding



membuat form validasi dengan php

dan nama pun berhasil di tampilkan sesuai dengan yang di input pada form nama. Sekian dulu tutorial Cara Membuat Form Validasi dengan PHP. teman-teman bisa mengembangkannya lagi sesuai dengan keperluan dan keinginan.

# Membuat Captcha dengan PHP

Di tutorial kali ini kita akan membuat sebuah captcha dengan menggunakan PHP. cara membuat captcha sangat lah mudah. kita hanya perlu membuat tulisan captcha dalam bentuk gambar agar tidak bisa di copas (Copy Paste). silahkan simak terlebih dulu penjelasan tentang captcha berikut ini untuk lebih mengenal captcha.

Sebelum lebih jauh tentang captcha, ada baiknya kita jelaskan dulu pengertian Captcha. karena mungkin ada teman-teman yang mencari dan belum tau pengertian captcha. Captcha adalah singkatan dari *Completely Automated Public Turing test to tell Computers and Humans Apart*. jadi dari kepanjangan captcha bisa kita ambil kesimpulan bahwa Captcha merupakan sebuah teknik ya bisa di katakan. sebuah teknik untuk mengecek robot atau manusia.

Captcha biasanya sebuah pengecekan atau pemeriksaan yang berbentuk gambar atau tulisan. dan biasanya captcha terdapat di sebuah form. jadi di sini captcha bertujuan untuk memeriksa bahwa yang mengisi form adalah manusia atau robot(komputer). bingung ? jangan bingung teman-teman. coba simak penjelasan lanjutnya.

Ada banyak orang yang membuat bot dengan tujuan sekedar mengisi form, melakukan spam melalui form dan ada juga dengan tujuan yang kurang baik yaitu melakukan brute force pada form login biasanya. nah tujuan dan kegunaan dari captcha ini adalah untuk menangkal bot yang mengisi form. jadi bisa menangkal spam pada form website kita. jadi captcha akan memeriksa yang mengisi form adalah manusia atau robot. kalau robot otomatis tidak akan bisa mengisi form karena ada captcha. bot tidak bisa mendeteksi tulisan apa yang terdapat dalam gambar captcha.

Oleh sebab itu captcha di buat dalam bentuk gambar. tetapi kalau manusia yang mengisi form nya pasti bisa karena manusia bisa membaca tulisan captcha dan kemudian bisa menuliskannya kembali untuk di validasi.

## Kegunaan captcha

Berikut kegunaan captcha yang saya sertakan dalam beberapa poin.

- Mencegah Spam
- Mencegah serangan brute force

Dan banyak lagi kegunaan dari captcha yang akan teman-teman rasakan sendiri setelah membuatnya.

## Cara Membuat Captcha dengan PHP

Sekarang kita akan langsung mulai membuat captcha sederhana menggunakan php. silahkan teman-teman simak penjelasan berikut ini tentang cara membuat captcha dengan mudah. kita buat dulu sebuah form. dimana nantinya pada form ini lah yang akan kita berikan captcha. buat sebuah file php dengan nama index.php di localhost teman-teman masing-masing.

index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Cara Membuat Captcha dengan php | WWW.MALASNGODING.COM</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <h1>Cara Membuat Captcha dengan php | WWW.MALASNGODING.COM</h1>
    <div class="kotak">

        <?php
        if(isset($_GET['pesan'])){
            if($_GET['pesan'] == "salah"){
                echo "<p>Captcha tidak sesuai.</p>";
            }
        }
        ?>

        <p>Isi Captcha Dengan Benar</p>
        <form action="periksa_captcha.php" method="post">
            <table align="center">
                <tr>
                    <td>Captcha</td>
                    <td> </td>
                </tr>
                <tr>
                    <td>Isikan captcha </td>
                    <td><input name="nilaiCaptcha" value="" /></td>
                </tr>
                <tr>
                    <td><input type ="submit" value="Cek Captcha"></td>
                </tr>
            </table>
        </form>
    </div>
</body>
</html>
```

lalu kita design dulu sedikit biar terlihat lebih rapi form nya.

### style.css

```
body{
    font-family: "roboto";
    background: #F4F4F4;
}

h1,p{
    text-align: center;
}

.kotak{
    margin: 10px auto;
    background: #fff;
    width: 400px;
    padding: 20px 0px;
}

.kotak table tr td{
    padding: 5px;
}

.kotak table tr td input{
    padding: 5px;
    font-size: 12pt;
}
```

buat sebuah file php lagi dengan nama captcha.php. file captcha.php ini lah yang akan membuat gambar captcha nya.

### captcha.php

```
<?php
//mengaktifkan session
session_start();

header("Content-type: image/png");

// menentukan session
$_SESSION["Captcha"]="";

// membuat gambar dengan menentukan ukuran
$gbr = imagecreate(200, 50);

//warna background captcha
imagecolorallocate($gbr, 69, 179, 157);

// pengaturan font captcha
```

```

$color = imagecolorallocate($gbr, 253, 252, 252);
$font = "Allura-Regular.otf";
$sukuran_font = 20;
$posisi = 32;

// membuat nomor acak dan ditampilkan pada gambar
for($i=0;$i<=5;$i++) {
    // jumlah karakter
    $angka=rand(0, 9);
    $_SESSION["Captcha"].=$angka;
    $kemiringan= rand(20, 20);
    imagettfttext($gbr, $sukuran_font, $kemiringan, 8+15*$i, $posisi, $color,
    $font, $angka);
}

//untuk membuat gambar
imagepng($gbr);
imagedestroy($gbr);
?>

```

untuk membuat captcha kita memerlukan session php. jadi kita aktifkan dulu session php.

```
session_start();
```

kemudian memberikan fungsi berikut untuk dapat membuat gambar dengan php.

```
header("Content-type: image/png");
```

dan selanjutnya syntax berikutlah yang bertugas untuk membuat gambar dengan php. silahkan teman-teman pahami.

```

// menentukan session
$_SESSION["Captcha"]="";
// membuat gambar dengan menentukan ukuran
$gbr = imagecreate(200, 50);

//warna background captcha
imagecolorallocate($gbr, 69, 179, 157);

// pengaturan font captcha
$color = imagecolorallocate($gbr, 253, 252, 252);
$font = "Allura-Regular.otf";
$sukuran_font = 20;
$posisi = 32;

// membuat nomor acak dan ditampilkan pada gambar
for($i=0;$i<=5;$i++) {
    // jumlah karakter

```

```
$angka=rand(0, 9);
$_SESSION["Captcha"].=$angka;
$kemiringan= rand(20, 20);
imagettftext($gbr, $ukuran_font, $kemiringan, 8+15*$i, $posisi, $color,
$font, $angka);
}
//untuk membuat gambar
imagepng($gbr);
imagedestroy($gbr);
```

jadi angka captcha nya kita buat secara random seperti yang teman-teman lihat di atas.

```
for($i=0;$i<=5;$i++) {
    // jumlah karakter
    $angka=rand(0, 9);
    $_SESSION["Captcha"].=$angka;
    $kemiringan= rand(20, 20);
    imagettftext($gbr, $ukuran_font, $kemiringan, 8+15*$i, $posisi, $color,
$font, $angka);
}
```

memberikan text pada gambar captcha dengan pengaturan yang sudah saya pisah-pisah ke dalam variabel masing-masing agar teman-teman bisa lebih mudah memahaminya.

```
imagettftext($gbr, $ukuran_font, $kemiringan, 8+15*$i, $posisi, $color,
$font, $angka);
```

oh ya jangan lupa untuk menyertakan file font yang akan teman-teman gunakan. perhatikan pada variabel font di atas. saya menggunakan font allura-regular.otf. teman-teman bisa mendownload font yang di sukai <https://www.1001freefonts.com/>.

```
$font = "Allura-Regular.otf";
```

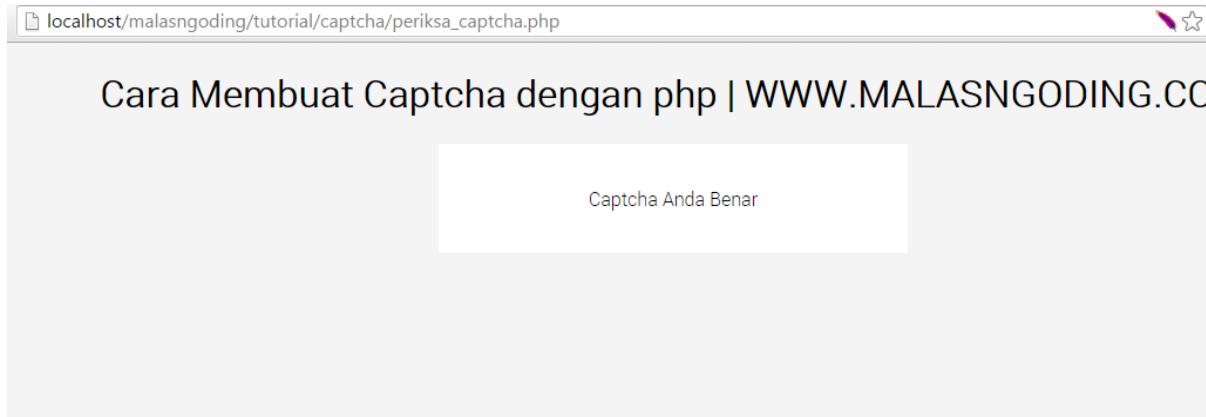
Disk (C:) ▶ xampp ▶ htdocs ▶ malasngoding ▶ tutorial ▶ captcha			
Share with ▾	Burn	New folder	
Name	Date modified	Type	Size
Allura-Regular		OpenType font file	48 KB
captcha	2/14/2016 4:54 PM	PHP File	1 KB
index	2/14/2016 4:58 PM	PHP File	1 KB
periksa_captcha	2/14/2016 4:59 PM	PHP File	1 KB
style	2/14/2016 4:53 PM	CSS File	1 KB

cara membuat captcha dengan php

Oke sampai di sini akan langsung kita coba saja captcha hasil buatan kita. buka file index.php di localhost.

cara membuat captcha dengan php

Isikan captcha yang sesuai dengan gambar captcha yang muncul. jika captcha yang di masukkan benar maka akan muncul pemberitahuan bahwa captcha yang di masukkan benar. tetapi jika salah maka akan di munculkan pesan captcha yang di masukkan salah. klik cek captcha.



membuat captcha sederhana dengan mudah

Selesai sudah captcha yang kita buat. silahkan teman-teman kembangkan lagi sesuai dengan keinginan masing-masing.

# Membuat dan Menghapus Folder Dengan PHP

Dengan PHP kita dapat membuat folder atau menghapus folder. tutorial inilah yang akan kita bahas pada tutorial kali ini. di PHP sudah di sediakan fungsi/function khusus untuk membuat folder baru, atau menghapus folder baru. kedua fungsi ini bisa kita gunakan di php sesuai dengan keinginan kita. nanti di tutorial selanjutnya akan saya bahas juga tentang tutorial cara membuat dan menghapus file dengan php.

Untuk membuat folder baru atau menghapus sebuah folder di php kita bisa menggunakan fungsi berikut ini:

- `mkdir()` = untuk membuat sebuah folder atau directory.
- `rmdir()` = untuk menghapus folder atau directory.

## Kegunaan `mkdir()` untuk membuat folder atau directory baru

Kita mulai dengan membahas fungsi `mkdir()` untuk membuat folder atau directory baru. silahkan teman-teman perhatikan. untuk cara penggunaan fungsi `mkdir()` di php ini cara penulisannya sebagai berikut.

```
<?php  
//cara penggunaan fuction mkdir()  
mkdir('nama folder baru','aksi setelah membuat folder');  
?>
```

Bisa diperhatikan pada contoh di atas. untuk membuat folder atau directory baru dengan php. kita menggunakan fungsi `mkdir()`. lalu isikan nama folder/directory yang ingin di buat. dan pada parameter keduanya merupakan opsional. bisa di isi atau tidak. pada parameter kedua berfungsi untuk kelakukan atau behavior saat folder berhasil di buat.

```
<?php  
mkdir("malas ngoding");  
?>
```

Atau

```
<?php  
function hai(){  
echo "Folder berhasil di buat";  
}
```

```
mkdir("malas ngoding",hai());
?>
```

oke kita coba jalankan pada browser.

index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Cara Membuat dan Menghapus Folder Dengan PHP |  
WWW.MALASNGODING.COM</title>
</head>
<body>
    <h1>Cara Membuat dan Menghapus Folder Dengan PHP |  
WWW.MALASNGODING.COM</h1>
    <?php
        function hai(){
            echo "Folder berhasil di buat";
        }

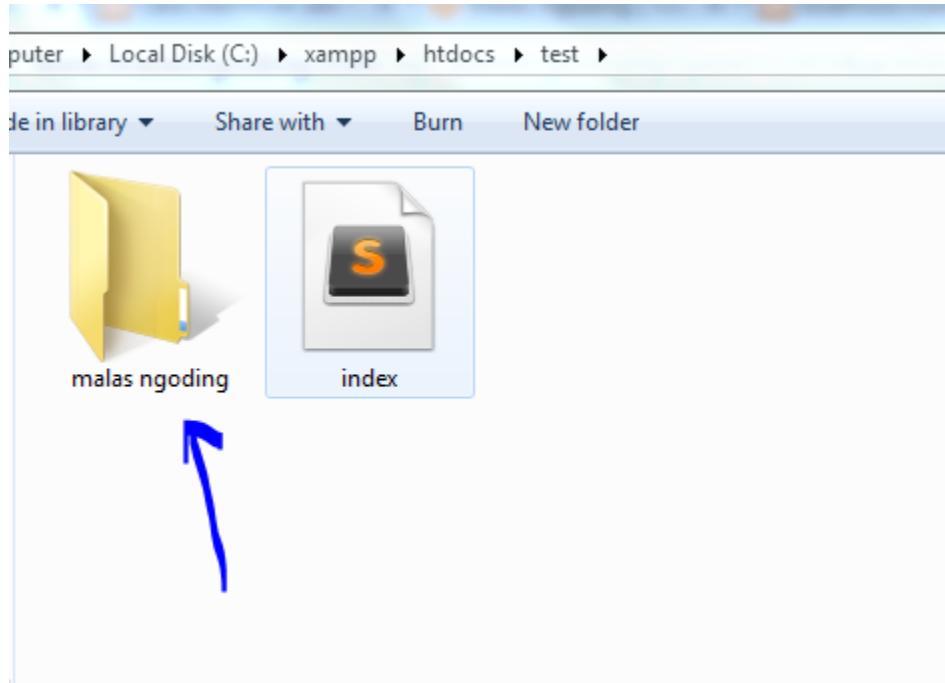
        mkdir("malas ngoding",hai());
    ?>
</body>
</html>
```



cara membuat dan menghapus folder atau directory di php

Coba teman-teman perhatikan pada contoh di atas. saya membuat function dengan nama hai. dan berisi perintah untuk menampilkan kata folder berhasil di buat. fungsi ini kita letakkan pada parameter kedua untuk menampilkan pesan bahwa folder berhasil di buat pada saat folder telah berhasil di buat.

Folder berhasil dibuat dengan php. Sekarang kita cek langsung apakah foldernya telah tersedia?



membuat folder atau directory dengan php

## Kegunaan rmdir() untuk menghapus folder atau directory

Setelah berhasil membuat folder dengan php sekarang kita akan mencoba menghapus folder dengan php. seperti yang telah di jelaskan di atas. untuk menghapus folder dengan php kita bisa menggunakan fungsi rmdir().

index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Cara Membuat dan Menghapus Folder Dengan PHP |  

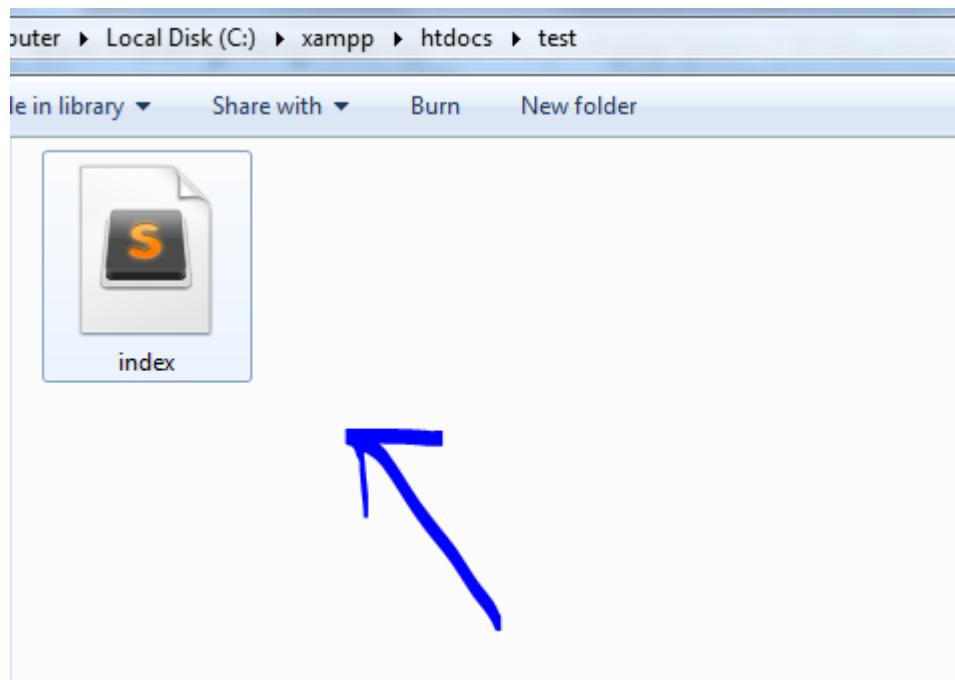
    WWW.MALASNGODING.COM</title>
</head>
<body>
    <h1>Cara Membuat dan Menghapus Folder Dengan PHP |  

    WWW.MALASNGODING.COM</h1>
    <?php  

        rmdir("malas ngoding");  

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

jalankan di browser dan lihat apakah folder berhasil di hapus.



menghapus folder atau directory dengan php

dan folder malas ngoding pun berhasil di hapus.

Oke sekian dulu tutorial Cara Membuat dan Menghapus Folder Dengan PHP.

# Mengenal MD5 Pada PHP

Pada tutorial mengenal MD5 Pada PHP ini. kita akan membahas tentang pengertian dari MD5, untuk apa saja kegunaannya dan bagaimana sih cara membuat MD5 pada PHP.

## Pengertian MD5

Menurut yang saya kutip dari wikipedia, MD5 adalah singkatan dari “*Message-Direct Algorithm 5*”. dan merupakan sebuah algoritma fungsi hash dari kriptografi yang di gunakan dengan hash value 128-bit.

MD5 telah di gunakan untuk berbagai macam keperluan dalam membangun keamanan sebuah sebuah aplikasi. khususnya MD5 Sering di gunakan untuk membantu validasi login pada sistem. penggunaan md5 dalam menyamarkan password yang tersimpan di database. agar menambah nilai secure pada sebuah sistem.

MD5 di kembangkan dan di rancang pada tahun 1991 oleh seorang yang bernama Ronald Rivest. untuk menggantikan fungsi hash sebelumnya, yaitu MD4.

Saya rasa untuk pengertian yang lebih lengkapnya lagi teman-teman bisa *googling* untuk mendapatkan informasi dari berbagai sumber.

kita lanjut ke kegunaannya pada php ya.

## Kegunaan MD5

Setelah kita mengenal sedikit pengertian dari MD5, sekarang kita akan membahas sedikit banyak nya kegunaan dari fungsi MD5 di dalam membangun keamanan sebuah aplikasi atau sistem informasi.

Biasanya MD5 di gunakan untuk meng-enkripsi data yang bersifat rahasia dan tidak ingin di ketahui oleh orang lain. contoh nya yang paling sering adalah dalam membuat sebuah login. biasanya data password pada database, akan di enkripsi terlebih dahulu. agar jika pun ada orang yang tidak bertanggung jawab yang masuk dan dapat melihat isi database, maka ia tidak akan bisa menebak password yang tersimpan di sana.(karena sudah di enkripsi dengan MD5).

MD5 sangat populer di kalangan back-end developer. untuk alasan keamanan.

Setelah ini, Kita juga akan membahas tutorial CARA MEMBUAT LOGIN DENGAN PHP, MYSQL DAN ENKRIPSI MD5. makanya tutorial ini di buat terlebih dulu agar memberikan sekilas bayangan terlebih dulu kepada teman-teman. agar lebih mudah nantinya mengikuti tutorial cara membuat login dengan bantuan md5.

seiring perjalanan teman-teman nantinya dalam mengembangkan aplikasi atau sistem informasi, teman-teman juga akan menemukan kegunaan lainnya dari fungsi MD5.

## Cara Membuat MD5 Pada PHP

PHP sudah menyediakan sebuah fungsi khusus untuk membuat atau untuk melakukan enkripsi MD5 dengan mudah. yaitu kita bisa menggunakan sebuah fungsi “md5()”.

contohnya :

Buat sebuah file php dengan nama apa saja. di sini saya membuat file php dengan nama enkripsi.php.

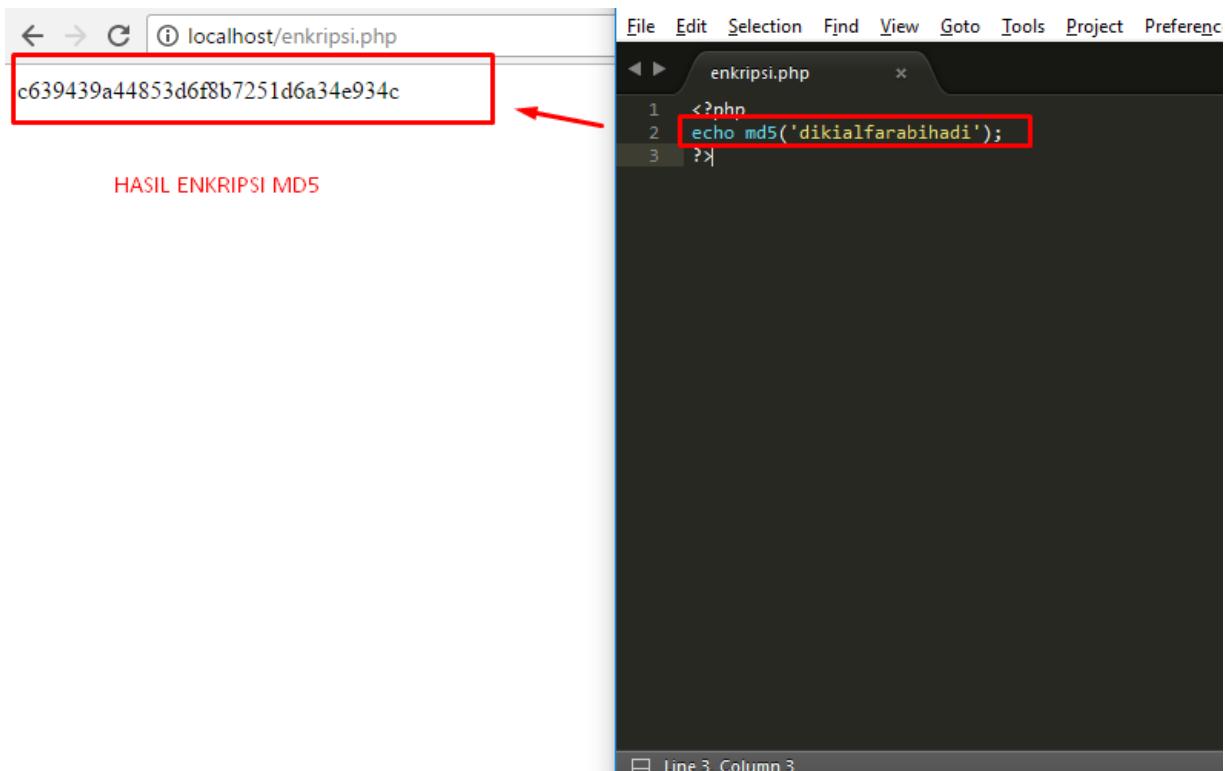
C:/xampp/htdocs/enkripsi.php

```
<?php  
echo md5('dikialfarabihadi');  
?>
```

Perhatikan cara penulisan dan penggunaan fungsi md5 pada PHP di atas. data yang ingin di enkripsi dengan md5 di tuliskan di dalam tanda kurung pembuka "(" dan kurung penutup ")".

jika kita jalankan, maka hasilnya

localhost/enkripsi.php



```
<?php  
echo md5('dikialfarabihadi');  
?>
```

pengertian penggunaan dan cara membuat md5 pada php

Seperti yang terlihat pada contoh di atas. "c639439a44853d6f8b7251d6a34e934c" adalah hasil enkripsi md5 dari "dikialfarabihadi".

contoh lain :

```
<?php  
echo md5('malasngoding');  
?>
```

Hasil yang tampil nantinya tidak lagi "malasngoding". tapi telah di samarkan dengan hasil enkripsi dari md5. Saya rasa cukup sekian dulu tutorial Pengertian, Penggunaan dan Cara Membuat MD5 Pada PHP.

# Membuat Koneksi Database PHP dan MySQL

Di tutorial kali ini kita akan belajar membuat koneksi php dengan database mysql, untuk menghubungkan php dengan database mysql. kenapa database mysql harus di hubungkan dengan php ? karena untuk memanipulasi database menggunakan php kita harus mengkoneksikan nya dulu. cara nya sangat mudah. PHP sudah menyediakan fungsi-fungsi khusus untuk menghubungkan php dengan database. yang akan kita jelaskan pada tutorial Cara Membuat Koneksi PHP Dengan Database MySQL.

## Mempersiapkan Database

Sebelum masuk ke tutorial menghubungkan php dengan database mysql. kita akan membuat sebuah database dulu untuk contoh saja. agar kita bisa belajar membuat koneksi database dengan mysql. karena jika database nya belum di buat jadi php nya mau di hubungkan kemana ? hehe. pertama buatlah sebuah database dengan nama 'malasngoding'. dan kemudian buat sebuah table dengan nama 'user'. isi column id, nama, alamat, pekerjaan pada table user. atau teman-teman bisa mengimport database nya langsung. berikut saya sediakan file sql nya untuk contoh saja. silahkan teman-teman import ke database yang bernama malasngoding yang sudah kita buat tadi.

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Feb 12, 2016 at 09:28 AM
-- Server version: 5.5.27
-- PHP Version: 5.4.7

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `malasngoding`
--
```

```

-- -----
-- 
-- Table structure for table `user` 

CREATE TABLE IF NOT EXISTS `user` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `pekerjaan` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=64 ; 

-- 
-- Dumping data for table `user` 

-- 

INSERT INTO `user` (`id`, `nama`, `alamat`, `pekerjaan`) VALUES
(1, 'Andi', 'Surabaya', 'web programmer'),
(2, 'Santoso', 'Jakarta', 'Web Designer'),
(6, 'Samsul', 'Sumedang', 'Pegawai');

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## Menghubungkan PHP Dengan Database MySQL

Setelah teman-teman selesai membuat database seperti yang saya intruksikan di atas, selanjutnya baru kita mulai membuat koneksi database php dan mysql. caranya buatlah sebuah file php dengan nama nya terserah teman-teman. di sini saya membuat file dengan nama `koneksi.php`. simpan file php tersebut pada directory localhost anda.

### `koneksi.php`

```

<?php
// isi nama host, username mysql, dan password mysql anda
mysql_connect("localhost","root","xxx");

// isikan dengan nama database yang akan di hubungkan
mysql_select_db("malasngoding");
?>
```

Seperti penjelasan di atas. pada function `mysql_connect()` isikan nama host anda, username mysql, dan password mysql anda. password mysql saya di sini adalah

xxx. jadi saya isi dengan xxx. dan username mysql saya root(default). dan nama host localhost. karena saya menggunakan localhost. mysql\_connect adalah fungsi php untuk menghubungkan file php dengan host. dan memberikan settingan nama host, username dan password mysql.

Fungsi mysql\_select\_db() adalah fungsi yang di sediakan php untuk memilih database yang ingin di hubungkan. jadi teman-teman tinggal mengisikan saya nama database yang akan di gunakan pada parameter fungsi mysql\_select\_db() seperti contoh di atas. sampai di atas kita sudah selesai membuat koneksi database php dan mysql. untuk menggunakannya teman-teman hanya tinggal menyisipkan file koneksi ini ke project anda. untuk menyisipkan nya teman-teman bisa menggunakan fungsi include(). baca tutorial saya sebelumnya tentang mengenal include dan require pada php.

Tetapi sampai di sini kita belum tau nih file koneksi database yang udah kita buat di atas ini udah betul atau berhasil belum. oleh karena itu kita bisa memeriksa koneksi nya dengan cara mengguakan percabangan kondisi if else pada php.

### koneksi.php

```
<?php
// isi nama host, username mysql, dan password mysql anda
$host = mysql_connect("localhost","root","xxx");

if($host){
    echo "koneksi host berhasil.<br/>";
} else{
    echo "koneksi gagal.<br/>";
}
// isikan dengan nama database yang akan di hubungkan
$db = mysql_select_db("malasngoding");

if($db){
    echo "koneksi database berhasil.";
} else{
    echo "koneksi database gagal.";
}
?>
```

dan jalankan file koneksi.php. jika koneksi berhasil maka akan di tampilkan koneksi host berhasil. tetapi jika gagal atau ada kesalahan dalam mengisi username atau password akan di tampilkan koneksi gagal. begitu juga dengan nama database. jika

database yang di tuju tersedia maka akan di tampilkan koneksi database berhasil. jika tidak akan di tampilkan koneksi database gagal.



koneksi host berhasil.  
koneksi database berhasil.

menghubungkan php dengan mysql

Sekianlah tutorial Cara Membuat Koneksi PHP Dengan Database MySQL.

# Membuat Login dengan php dan mysql menggunakan enkripsi md5

Login adalah proses yang paling penting dan paling di utama kan pada sebuah website, aplikasi yang berbasis website maupun aplikasi lainnya yang membatasi akses oleh user agar tidak dapat mengakses bagian dalam sistem misalnya seperti bagian dari admin. oleh karena itulah login di gunakan sebagai pengaman dari data-data atau masing-masing akun.

Jika sebuah website tidak memiliki proses login maka semua orang akan dapat mengakses halaman admin dan dapat dengan bebas untuk ngupdate postingan, menghapus postingan atau bebas ngacak-ngacak isi dari website. dan contoh lainnya coba bayangkan bagaimana jadi nya kalau akun facebook kita dapat di akses oleh semua orang tanpa harus login. oleh sebab itu login sangat penting pada website maupun semua aplikasi untuk menjaga keamanan dari akses luar.

Untuk lebih mengamankan proses login kita akan menambahkan enkripsi md5 pada password untuk lebih memberikan keamanan yang lebih. jadi misalnya ada yang mengambil database dan table yang menyimpan data username dan password anda maka agak lebih sedikit aman karena akan sulit untuk mengetahui password yang sudah di enkripsi dengan md5.

oke langsung saja kita akan belajar cara membuat form login dengan php dan mysql secara bertahap step by step agar lebih di pahami oleh anda yang baru mulai belajar membuat login dengan php, mysql dan bantuan enkripsi md5 untuk keamanan enkripsi password nya.

## Persiapan Database dan Table

Baiklah di sini saya akan jelaskan tentang cara membuat form login standar dulu ya tanpa menggunakan enkripsi md5, maksud standar yang saya maksud ini iyalah tidak menggunakan enkripsi md5. agar lebih mudah teman-teman pahami. dan pada tahap selanjutnya pada tutorial ini juga akan saya bahas cara membuat form login dengan php dan enkripsi md5.

Baiklah untuk persiapan di sini sudah saya siapkan database dengan nama "*malasngoding*". dan sebuah table yang berisi tiga colum yaitu id,username dan

password. data username admin yang akan login kita simpan pada username dan data password kita simpan pada password. silahkan anda import sendiri ini table adminnya. jangan lupa buat database dengan nama malasngoding dulu dan kemudian baru import table admin berikut ini.

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Jan 23, 2016 at 05:42 AM
-- Server version: 5.5.27
-- PHP Version: 5.4.7

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `malasngoding`
--


-----
-- Table structure for table `admin`


CREATE TABLE IF NOT EXISTS `admin` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

--
-- Dumping data for table `admin`
--


INSERT INTO `admin` (`id`, `username`, `password`) VALUES
(1, 'malasngoding', 'malasngoding123');

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

setelah selesai di import

The screenshot shows the MySQL Workbench interface. At the top, the connection path is 127.0.0.1 » malasngoding » admin. Below the connection path are several tabs: Browse, Structure, SQL, Search, Insert, Export, and Import. The SQL tab is active. A green status bar at the top indicates "Showing rows 0 - 0 (~1 total) , Query took 0.0005 sec". In the main area, there is a code editor with the following SQL query:

```
SELECT *  
FROM `admin`  
LIMIT 0, 30
```

Below the code editor is a search bar with the placeholder "Search" and a magnifying glass icon. Underneath the search bar are filtering options: "Show : Start row: 0 Number of rows: 30 Headers every 100 rows". The main content area displays a table named "admin". The table has columns: id, username, and password. There is one row with the values: id=1, username=malasngoding, and password=malasngoding123. Below the table are buttons for Edit, Copy, Delete, Change, Delete, and Export.

Sampai pada tahap ini berarti database dan table admin sudah siap untuk di gunakan membuat form login dengan php. langkah selanjutnya kita akan membuat tampilan form login sederhana.

## Membuat Form Login Sederhana

Buat folder baru untuk project baru ini. di sini saya membuat folder dengan nama "*login*" di localhost untuk membuat contoh cara membuat login dengan php dan mysql ini. selanjutnya buat sebuah file php untuk kita buat menjadi form loginnya. di sini saya membuat file php nya dengan nama index.php dalam folder "*login*". karena saya ingin form login ini tampil yang pertama saat di akses.

### index.php

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Membuat Login Dengan PHP dan MySQL | MalasNgoding.com</title>  
</head>
```

```

<body>
    <h1>Membuat Login Dengan PHP dan MySQL | MalasNgoding.com</h1>
    <h3>Halaman Login Sederhana</h3>
    <form action="login.php" method="post">
        <table>
            <tr>
                <td>Username</td>
                <td><input type="text" name="username"></td>
            </tr>
            <tr>
                <td>Password</td>
                <td><input type="password" name="password"></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" name="login" value="Log In"></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

untuk tampilan form login ini kita buat yang paling sederhana saja dulu. karena tutorial ini kita fokuskan untuk pembelajaran tentang cara membuat form login dengan php dan mysql saja. untuk tutorial design form akan kita bahas juga pada tutorial kategori web design di malasngoding.com.



## Membuat Login Dengan PHP dan MySQL | MalasNgoding.com

### Halaman Login Sederhana

Username	<input type="text"/>
Password	<input type="password"/>
	<input type="button" value="Log In"/>

membuat halaman login dengan php

kemudian yang tidak kalah pentingnya adalah buat sebuah file koneksi.php untuk menghubungkan project tutorial membuat login dengan php kita ini dengan database.

## koneksi.php

```
<?php  
mysql_connect("localhost","root","xxx"); //sesuaikan dengan password dan  
username mysql anda  
mysql_select_db("malasngoding"); //nama database yang kita gunakan  
?>
```

Perhatikan pada form login yang sudah kita buat, kita menentukan aksi dari form tersebut ke file login.php. sehingga secara otomatis halaman akan di alihkan ke halaman login.php pada saat tombol login di klik atau form di submit. jadi pada file login.php ini lah kita membuat aksi dari form, mulai dari menangkap data username dan password yang di ketikkan pada form sebelum di submit.

kemudian mencocokkan username dan password yang di ketik dengan username dan password yang ada pada database. jika username dan password yang di ketik pada form login sesuai dengan yang ada di database berarti login sukses tetapi jika username dan password yang di input pada form login tidak sesuai dengan yang ada di database berarti login gagal karena username dan password yang di masukkan salah.

## login.php

```
<?php  
include 'koneksi.php';  
  
$username = $_POST['username'];  
$password = $_POST['password'];  
  
$query = mysql_query("select * from admin where username='$username' and  
password='$password'");  
$cek = mysql_num_rows($query);  
echo $cek;  
?>
```

Perhatikan login.php di atas. pertama kita hubungkan dulu dengan koneksi.php dengan menggunakan fungsi include atau require, di sini saya menggunakan include saja. include dan require sudah saya jelaskan pada tutorial php sebelumnya di malasngoding.com.

Dan kemudian menangkap data username dan password yang di input pada form login lalu memasukkannya ke dalam variabel

```
$username = $_POST['username'];
$password = $_POST['password'];
```

Kemudian lagi kita menggunakan fungsi *mysql\_query()* untuk menjalankan syntax atau perintah mysql pada php. jadi *mysql\_query()* adalah sebuah fungsi php yang berguna untuk menjalankan perintah mysql. perintah mysql dapat anda letakkan pada parameter fungsi *mysql\_query()* seperti yang dapat teman-teman lihat pada contoh di atas.

```
$query = mysql_query("select * from admin where username='$username' and password='$password');

$cek = mysql_num_rows($query);
echo $cek;
```

di sini kita menjalankan perintah mysql untuk memilih atau mengambil data username yang sama dengan username yang di input dan password yang di input dari table admin. kemudian hasil query mysql ini kita hitung jumlah record yang di temukan dengan fungsi *mysql\_num\_rows()*. fungsi *mysql\_num\_rows()* adalah fungsi php yang berguna untuk menghitung jumlah record yang di temukan dari query mysql seperti yang dapat teman-teman sekalian lihat pada contoh di atas. nah, jadi apabila username dan password yang di input ada pada database maka nilai variabel \$cek di atas akan berisi 1 dan jika tidak ada maka nilai pada variabel \$cek adalah 0.

Dengan 1 dan 0 inilah yang akan kita jadikan pendekripsi apakah username dan password yang di input pada form login benar atau salah. jika benar berarti nilainya adalah 1 dan jika salah nilainya adalah 0. pada contoh ini saya menampilkan isi nilai dari variabel \$cek. tadi pada table admin kan berisi username dengan nama malasngoding dan passwordnya adalah malasngoding123. sekarang kita tes ya dengan memasukkan username dan password yang salah.



## Membuat Login Dengan PHP dan MySQL | MalasNgoding.com

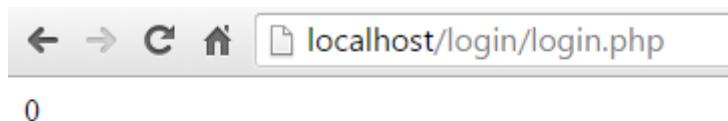
### Halaman Login Sederhana

Username

Password

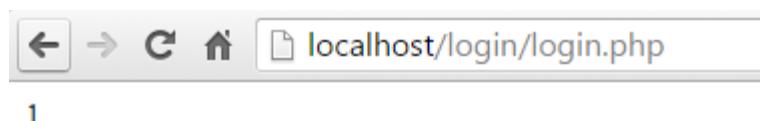
memasukkan username dan password yang salah

Untuk mengetes memasukkan username dan password yang salah saya memasukkan username dengan nama “andi” dan password nya “andi123”. dan saat di klik login maka nilai yang muncul adalah 0, karena username “andi” dan password “andi123” tidak di temukan pada table admin kita.



login php gagal

dan bagaimana jika kita memasukkan username dengan “malasngoding” dan passwordnya “malasngoding123”. maka akan muncul hasil nya 1 karena username malasngoding dan password malasngoding123 tersedia pada table admin kita.



login php berhasil

Nah muncul nilai 1 jika kita memasukkan username malasngoding dan password malasngoding123 pada form login. sampai di sini kita telah berhasil membuat form login yang sudah mampu mendeteksi username dan password yang salah atau benar sesuai dengan username dan password yang terdapat pada database.

Apakah sudah selesai sampai di sini form login php kita ? tentu saja belum. selanjutnya kita akan membuat aksi pada jika username dan password yang di masukkan benar atau salah. jika username dan password yang di masukkan salah maka akan kita alihkan kembali ke halaman login. tetapi jika username dan password yang di masukkan benar maka akan kita buat session untuk si admin yang melakukan login. nah apa itu session ? session akan kita gunakan sebagai penanda apakah si admin yang melakukan login sudah dalam posisi berhasil login atau belum.

Jadi dengan session ini lah yang akan membantu kita dalam pemeriksaan user atau admin yang sudah login atau belum. jadi yang belum login akan kita alihkan pada halaman login dan yang sudah login langsung di alihkan ke halaman admin. jadi session php sangat berguna dalam proses membuat login.

Selanjutnya akan kita lanjutkan dengan membuat aksi dari pengecekan login. jadi di sini kita akan membuat session pada saat username dan password yang di masukkan benar dan membuat pengalihan halaman ke halaman login kembali jika username dan password yang di masukkan salah.

## Membuat validasi login dengan PHP dan mysql

Buka file login.php yang sudah dibuat sebelumnya. Dan perhatikan penjelasan berikut.

### login.php

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

$username = $_POST['username'];
$password = $_POST['password'];

$query = mysql_query("select * from admin where username='$username' and
password='$password'");
$cek = mysql_num_rows($query);

if($cek > 0){
    session_start();
    $SESSION['username'] = $username;
```

```
$_SESSION['status'] = 'login';
header("location:admin/index.php");
} else{
    header("location:index.php");
}
?>
```

Perhatikan pada syntax di atas, kita membuat validasi dengan mengecek nilai pada variabel \$cek. jika nilai lebih besar dari nol (berarti username dan password benar/data login tersedia dan sesuai) maka kita membuat session.

```
if($cek > 0){
    session_start();
    $_SESSION['username'] = $username;
    $_SESSION['status'] = 'login';
    header("location:admin/index.php");
} else{
    header("location:index.php");
}
```

Sedikit penjelasan di sini. setiap penggunaan session php anda harus menjalankan fungsi `session_start()` terlebih dahulu untuk memulai session. setelah session di start baru penggunaan session lainnya di php bisa di lakukan. seperti yang dapat anda perhatikan pada syntax di atas. jika username dan password yang di masukkan benar maka kita membuat session username yang berisi username user/admin yang login. dan juga membuat session dengan nama status yang berisi 'login'. kemudian mengalihkan halaman ke halaman admin. halaman admin yang saya buat di sini terletak pada folder admin. oleh karena itu kita buat dulu folder admin. dalam folder admin ini kita akan membuat halaman sederhana untuk admin. jadi jika login berhasil maka halaman akan di alihkan pada halaman ini.

Buat sebuah folder dengan nama “*admin*” dan buat sebuah file baru dengan nama index.php di dalam nya

admin/index.php

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
</head>
```

```
<body>
<?php include 'cek-login.php'; ?>
<h1>HALAMAN ADMIN</h1>

<h3><?php echo "Selamat Datang, Pak ".$_SESSION['username']; ?></h3>

<a href="logout.php">Logout</a>
</body>
</html>
```

Buat juga file php dengan nama *cek-login.php*. file cek-login.php ini kita gunakan sebagai pendekripsi apakah user telah login atau belum. jika belum login maka admin/user tidak bisa mengakses halaman admin ini(di alihkan ke halaman login).

admin/cek-login.php

```
<?php
session_start();
if($_SESSION['status'] == ""){
    header("location:../index.php");
}
?>
```

dapat di perhatikan pada file cek-login.php di atas. karena di sini kita jika akan menggunakan session maka tetap kita harus memulai session dulu dengan fungsi session\_start(). kemudian melakukan sessionn

```
if($_SESSION['status'] == ""){
    header("location:../index.php");
}
```

Maksudnya gini. tadi pada saat login berhasil kan otomatisi akan di buat session dengan nama username dan status. nah di sini kita cek apakah session status kosong? kalau kosong berarti kan user/admin belum login, jadi tidak kita perbolehkan untuk mengakses halaman admin ini. jadi halaman akan kita alihkan ke halaman index.php atau halaman login.

Selanjutnya kita buat file dengan nama logout.php dalam folder admin juga.

admin/logout.php

```
<?php  
session_start();  
session_destroy();  
header("location:../index.php");  
?>
```

Pertama tetap memulai session dulu. kemudian `session_destroy()` berfungsi untuk menghapus seluruh session. kemudian halaman akan di alihkan ke halaman index.php (halaman login).

Dan selesai. sekarang waktu nya kita coba dulu nih form login yang telah di buat. coba akses langsung ke halaman admin tanpa melakukan login. maka secara otomatis halaman akan di alihkan ke halaman login kembali karena kita belum melakukan login. dan kemudian coba juga untuk memasukkan username dan password yang benar maka akan di alihkan ke halaman admin. dan klik pada link logout untuk menghapus semua session dan logout.



localhost/login/admin/index.php

# HALAMAN ADMIN

**Selamat Datang, Pak malasngoding**

[Logout](#)

membuat login dengan php dan mysql

dan selesai.

## Login Dengan Enkripsi Password menjadi md5

Untuk membuat enkripsi md5, PHP telah menyediakan fungsi yang bisa langsung di gunakan untuk membuat enkripsi md5. yaitu dengan menggunakan fungsi md5(). kata yang ingin di enkripsi bisa di letakkan pada parameter fungsi md5() misalnya md5(' malasngoding ') dan lainnya.

```
echo md5('malasngoding');
```

Jalankan syntax di atas untuk mendapatkan hasil enkripsi md5 'malasngoding'.

Syntax di atas hanyalah contoh penggunaan fungsi md5 pada php. jadi untuk membuat login dengan enkripsi password silahkan perhatikan penjelasan berikut ini.

kita dapatkan dulu hasil enkripsi md5 password kita. dengan menjalankan syntax echo md5('malasngoding123'); dan hasilnya adalah 10406c1d7b7421b1a56f0d951e952a95. jadi masukkan 10406c1d7b7421b1a56f0d951e952a95 ke column password pada table admin. kemudian buka file login.php. dan berikan fungsi md5() pada variabel \$password.

```
$password = md5($_POST['password']);
```

sehingga menjadi

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

$username = $_POST['username'];
$password = md5($_POST['password']);

$query = mysql_query("select * from admin where username='$username' and
password='$password'");

$cek = mysql_num_rows($query);

if($cek > 0){
    session_start();
    $_SESSION['username'] = $username;
    $_SESSION['status'] = 'login';
    header("location:admin/index.php");
} else{
    header("location:index.php");
}
?>
```

Jadi data inputan yang di tangkap kita jadikan enkripsi md5. dan kemudian hasil enkripsinya yang kita **samakan** untuk pengecekan login username dan password.

sekian lah tutorial cara membuat login dengan PHP dan MySQL menggunakan MD5.

# Membuat CRUD Dengan PHP dan MySQL

Tutorial ini termasuk tutorial yang paling penting yang harus teman-teman pelajari. kenapa tidak di tutorial ini akan di jelaskan tentang pengertian CRUD, cara membuat CRUD, dan kenapa saya bilang tutorial crud ini yang paling terpenting ? karena crud di gunakan untuk membuat semua aplikasi. karena berhubungan dengan menginput data ke database, mengedit data dari database , menampilkan data dari database serta menghapus data yang tidak di inginkan pada database dengan menggunakan PHP.

Pasti sudah terbayang kan oleh teman-teman betapa pentingnya CRUD. teman-teman tidak akan bisa membuat sebuah aplikasi bahkan yang sangat sederhana sekalipun jika tidak menguasai konsep CRUD. kecuali aplikasi yang teman-teman buat tidak berhubungan dengan database. tutorial ini saya tujuhan kepada teman-teman yang mungkin baru belajar php. mungkin sudah mempelajari php dasar dan bingung ingin melanjutkan pembelajaran selanjutnya kemana. saya rekomendasikan kepada teman-teman untuk mempelajari tutorial php dasar terlebih dulu di pembahasan pada sub judul sebelumnya dalam ebook ini.

## Pengertian CRUD (Create, Read, Update dan Delete)

Sebelum memulai belajar cara membuat crud dengan php dan mysql dan sebelum lebih jauh lagi mengotak-atik . ada baik nya terlebih dulu saya jelaskan tentang pengertian dari CRUD karena mungkin masih banyak teman-teman yang masih bingung tentang apa sih crud ini?

CRUD adalah singkatan dari *Create, Read, Update and Delete*. jadi di singkat dengan CRUD. berbicara tentang create, read, update, delete. pasti kita terbayang tentang sebuah manajemen atau mengelola. yak benar sekali. CRUD di sini adalah pengelolaan database. misalnya seperti menginput data ke database (Create), Menampilkan data dari database (Read), Mengubah atau mengupdate data pada database (Update) dan menghapus data pada database(Delete).

Nah kesimpulan nya kita akan belajar tentang membuat form input yang menginput data ke database. cara menampilkan data dari database. mengubah data dari database serta belajar cara menghapus data pada database dengan php.

## Persiapan & Membuat Koneksi Database

Karena kita akan bermain dengan database, maka yang menjadi langkah awal kita adalah mempersiapkan database. Buatlah sebuah database dengan nama '*malasngoding*'. dan buat sebuah table bernama '*user*'. yang berisi column id, nama, alamat, pekerjaan.

Jadi di sini kita akan membuat crud untuk data user. Setelah teman-teman membuat database dengan nama '*malasngoding*', buat sebuah table dengan nama '*user*'. Dengan Column pada table user nya ( id, nama, alamat, pekerjaan ).

Berikut ini sudah saya sertakan file sql yang bisa teman-teman langsung import. cara mengimport nya buat databasenya dulu dengan nama *malasngoding*. kemudian pilih import untuk mengimport file sql nya. tapi jika teman-teman ingin copy paste sql nya bisa juga. pilih pada sql. kemudian pastekan sql berikut ini.

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Feb 12, 2016 at 09:28 AM
-- Server version: 5.5.27
-- PHP Version: 5.4.7

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--

-- Database: `malasngoding`
--


-----
-- Table structure for table `user`


CREATE TABLE IF NOT EXISTS `user` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `pekerjaan` varchar(50) NOT NULL,
```

```

PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=64 ;

-- 
-- Dumping data for table `user`
-- 

INSERT INTO `user` (`id`, `nama`, `alamat`, `pekerjaan`) VALUES
(1, 'Andi', 'Surabaya', 'web programmer'),
(2, 'Santoso', 'Jakarta', 'Web Designer'),
(6, 'Samsul', 'Sumedang', 'Pegawai');

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Connection:** 127.0.0.1 » malasngoding » user
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Operations, Trace.
- Status Bar:** Showing rows 0 - 2 (~3 total), Query took 0.0026 sec.
- SQL Editor:** SELECT \* FROM `user` LIMIT 0 , 30
- Table Data:** A grid showing the following data:

	id	nama	alamat	pekerjaan
<input type="checkbox"/>	1	Andi	Surabaya	web programmer
<input type="checkbox"/>	2	Santoso	Jakarta	Web Designer
<input type="checkbox"/>	6	Samsul	Sumedang	Pegawai

- Buttons:** Check All / Uncheck All With selected, Change, Delete, Export.

membuat database dan table user

Sampai di sini persiapan database dan tabel user telah selesai.

Setelah selesai membuat database, selanjutnya kita langsung masuk ke tahap membuat koneksi database dengan php dan mysql. oh iya pertama teman-teman harus paham dulu tentang membuat koneksi database mysql dan php. silahkan teman-

teman baca dulu tutorial pada sub judul sebelumnya tentang membuat koneksi database dengan PHP dan MySQL.

Pertama kita buat dulu sebuah file php dengan nama nya terserah teman-teman. di sini saya membuat sebuah file koneksi.php. di file koneksi.php inilah nantinya akan kita buat koneksi php dan mysql.

### koneksi.php

```
<?php  
// isi nama host, username mysql, dan password mysql anda  
$host = mysql_connect("localhost","root","xxx");  
  
// isikan dengan nama database yang akan di hubungkan  
$db = mysql_select_db("malasngoding");  
?>
```

silahkan teman-teman sesuaikan dengan username dan password mysql teman-teman. username saya root dan passwordnya xxx. jadi pengaturannya seperti di atas. Silahkan sesuaikan sendiri. Biasanya jika teman-teman tidak menggunakan password mysql, bisa di kosongkan saja pada parameter passwordnya. Buat jadi

```
$host = mysql_connect("localhost","root","");

```

Oke, file koneksi.php selesai. Selanjutnya kita akan mulai menampilkan data dari database dengan PHP dan MySQL.

### Menampilkan Data Dari Database

Kemudian buat sebuah file index.php. di file index.php ini kita akan menampilkan data dari database mysql. tepatnya dari table user yang telah kita buat di atas.

### index.php

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Membuat CRUD Dengan PHP Dan MySQL - Menampilkan data dari  
    database</title>  
    <link rel="stylesheet" type="text/css" href="style.css">  
</head>  
<body>
```

```

<div class="judul">
    <h1>Membuat CRUD Dengan PHP Dan MySQL</h1>
    <h2>Menampilkan data dari database</h2>
    <h3>www.malasngoding.com</h3>
</div>
<br/>

<?php
if(isset($_GET['pesan'])) {
    $pesan = $_GET['pesan'];
    if($pesan == "input") {
        echo "Data berhasil di input.";
    } else if($pesan == "update") {
        echo "Data berhasil di update.";
    } else if($pesan == "hapus") {
        echo "Data berhasil di hapus.";
    }
}
?>
<br/>
<a class="tombol" href="input.php">+ Tambah Data Baru</a>

<h3>Data user</h3>
<table border="1" class="table">
    <tr>
        <th>No</th>
        <th>Nama</th>
        <th>Alamat</th>
        <th>Pekerjaan</th>
        <th>Opsi</th>
    </tr>
    <?php
    include "koneksi.php";
    $query_mysql = mysql_query("SELECT * FROM user") or
die(mysql_error());
    $nomor = 1;
    while($data = mysql_fetch_array($query_mysql)) {
    ?>
    <tr>
        <td><?php echo $nomor++; ?></td>
        <td><?php echo $data['nama']; ?></td>
        <td><?php echo $data['alamat']; ?></td>
        <td><?php echo $data['pekerjaan']; ?></td>
        <td>
            <a class="edit" href="edit.php?id=<?php echo
$data['id']; ?>">Edit</a> |
            <a class="hapus" href="hapus.php?id=<?php echo
$data['id']; ?>">Hapus</a>
        </td>
    </tr>
    <?php } ?>
</table>
</body>
</html>

pada syntax
<?php

```

```

if(isset($_GET['pesan'])) {
    $pesan = $_GET['pesan'];
    if($pesan == "input") {
        echo "Data berhasil di input.";
    }else if($pesan == "update") {
        echo "Data berhasil di update.";
    }else if($pesan == "hapus") {
        echo "Data berhasil di hapus.";
    }
}
?>

```

Jangan teman-teman hiraukan dulu. akan saya jelaskan pada bagian bawah nanti. karena syntax di atas hanya pengambilan data melalui url untuk membuat pesan notifikasi misalnya data telah berhasil di update atau di hapus.

yang perlu teman-teman perhatikan adalah pada syntax berikut pada file index.php.

```

<table border="1" class="table">
    <tr>
        <th>No</th>
        <th>Nama</th>
        <th>Alamat</th>
        <th>Pekerjaan</th>
        <th>Opsi</th>
    </tr>
    <?php
    include "koneksi.php";
    $query_mysql = mysql_query("SELECT * FROM user") or die(mysql_error());
    $nomor = 1;
    while($data = mysql_fetch_array($query_mysql)) {
    ?>
    <tr>
        <td><?php echo $nomor++; ?></td>
        <td><?php echo $data['nama']; ?></td>
        <td><?php echo $data['alamat']; ?></td>
        <td><?php echo $data['pekerjaan']; ?></td>
        <td>
            <a class="edit" href="edit.php?id=<?php echo $data['id']; ?>">Edit</a> |
            <a class="hapus" href="hapus.php?id=<?php echo $data['id']; ?>">Hapus</a>
        </td>
    </tr>
    <?php } ?>
</table>

```

Dapat anda lihat pada syntax di atas. kenapa saya menggunakan element table html di sini. karena kita akan menampilkan data nya di dalam table ini. lanjut lagi pada

```

<?php
include "koneksi.php";
$query_mysql = mysql_query("SELECT * FROM user") or die(mysql_error());
$nomor = 1;
while($data = mysql_fetch_array($query_mysql)) {
?
<tr>
    <td><?php echo $nomor++; ?></td>
    <td><?php echo $data['nama']; ?></td>
    <td><?php echo $data['alamat']; ?></td>
    <td><?php echo $data['pekerjaan']; ?></td>
    <td>
        <a class="edit" href="edit.php?id=<?php echo $data['id']; ?>">Edit</a>
    |
        <a class="hapus" href="hapus.php?id=<?php echo $data['id']; ?>">Hapus</a>
    </td>
</tr>
<?php } ?>

```

Coba teman-teman perhatikan pada syntax di atas. pertama kita harus menghubungkan file index.php ini dengan koneksi.php. atau menyisipkan file koneksi.php pada index.php ini. karena pada file index.php ini kita memerlukan koneksi ke database untuk mengambil data dari database dan kemudian menampilkannya pada table. kita menyisipkan file koneksi.php dengan fungsi include.

```
include "koneksi.php";
```

Dan bisa teman-teman lihat juga terdapat fungsi mysql\_query(). fungsi mysql\_query berfungsi untuk menjalankan perintah mysql pada php. jadi perintah mysql nya teman-teman isikan pada fungsi mysql\_query() seperti di bawah ini. dan perintah mysql yang kita jalankan adalah “*select \* from user*” yang berarti kita memerintahkan untuk menampilkan semua data pada table *user*. lalu fungsi *die()* di bawah ini adalah fungsi php untuk menampilkan pesan error jika fungsi sebelumnya mengalami kesalahan.

```
$query_mysql = mysql_query("SELECT * FROM user") or die(mysql_error());
```

Perintah query yang menampilkan data dari table *user* kita masukkan kedalam variabel *\$query\_mysql*. jadi sekarang variabel *query\_mysql* di atas sudah berisi data *user*.

Selanjutnya bagaimana cara menampilkan data nya?

Kita akan menggunakan perulangan while php.

```
while($data = mysql_fetch_array($query_mysql)) {  
?>  
    <tr>  
        <td><?php echo $nomor++; ?></td>  
        <td><?php echo $data['nama']; ?></td>  
        <td><?php echo $data['alamat']; ?></td>  
        <td><?php echo $data['pekerjaan']; ?></td>  
        <td>  
            <a class="edit" href="edit.php?id=<?php echo $data['id']; ?>">Edit</a> |  
            <a class="hapus" href="hapus.php?id=<?php echo $data['id']; ?>">Hapus</a>  
        </td>  
    </tr>  
</?php } ?>
```

Pada fungsi while kita memecahkan data menjadi array dan memasukkannya ke dalam variabel data. agar data bisa kita tampilkan dalam bentuk perulangan.

```
$data = mysql_fetch_array($query_mysql)
```

lalu menampilkan data nya tinggal

```
<td><?php echo $data['nama']; ?></td>  
<td><?php echo $data['alamat']; ?></td>  
<td><?php echo $data['pekerjaan']; ?></td>
```

Variabel data sudah menjadi array. dan masukkan nama column data yang ingin di tampilkan seperti contoh di atas. juga ada variabel nomor di atas. kenapa saya membuat variabel nomor dengan tanda plus? karena di atas nilai \$nomor=1. dan memberikan ++ di dalam while akan membuat nilai nomor bertambah terus menerus terus menerus sesuai jumlah record data yang di tampilkan.

Terakhir perhatikan pada link edit dan hapus berikut. ini untuk membuat link edit data dan hapus data. data di edit atau di hapus dengan menggunakan pengiriman data melalui url(GET). untuk memahami GET dan POST di PHP silahkan teman-teman baca tutorial saya sebelumnya tentang method GET dan POST pada pembahasan sebelumnya.

```
<a class="edit" href="edit.php?id=<?php echo $data['id']; ?>">Edit</a> |  
<a class="hapus" href="hapus.php?id=<?php echo $data['id']; ?>">Hapus</a>
```

dan jangan lupa kita percantik tampilannya dengan CSS agar enak di pandang mata.

### style.css

```
body{
    font-family: 'roboto';
    color: #000;
}

.judul{
    background: #87D1D8;
    padding: 10px;
    text-align: center;
}

.judul h1,h2,h3{
    height: 15px;
}

a{
    /*color: #fff;*/
    padding: 5px;
    text-decoration: none;
}

.table{
    border-collapse: collapse;
}

table.table th th , table.table tr td{
    padding: 10px 20px      ;
}
```

Setelah penjelasan selesai langsung saja kita coba jalankan file index.php untuk melihat hasil menampilkan data dari database dengan php.

Membuat CRUD Dengan PHP Dan MySQL  
Menampilkan data dari database  
www.malasngoding.com

+ Tambah Data Baru

#### Data user

No	Nama	Alamat	Pekerjaan	Opsi
1	Andi	Surabaya	web programmer	Edit   Hapus
2	Santoso	Jakarta	Web Designer	Edit   Hapus
3	Samsul	Sumedang	Pegawai	Edit   Hapus

#### Data User Tampil Dari Database

Nah data pun berhasil kita tampilkan. Selesailah sudah proses menampilkan data dari database. selanjutnya akan kita bahas tentang membuat input data ke database dengan PHP pada sub judul selanjutnya.

### Input Data Ke Database

Pada sub judul sebelumnya kita telah membahas tentang tutorial cara menampilkan data dari database. maka pada tutorial ini saya akan menjelaskan tentang cara membuat insert/input data ke database.

Baiklah langsung saja kita lanjutkan tutorial tentang cara membuat crud dengan php dan mysql menginput data ke database. siapkan sebuah file dengan nama *input.php*. file *input.php* ini akan kita jadikan sebagai form inputan data. jadi data yang akan di insert ke database akan kita masukkan melalui form input di *input.php* ini.

#### input.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat CRUD Dengan PHP Dan MySQL – Menampilkan data dari
database</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

```

<body>
    <div class="judul">
        <h1>Membuat CRUD Dengan PHP Dan MySQL</h1>
        <h2>Menampilkan data dari database</h2>
        <h3>www.malasngoding.com</h3>
    </div>

    <br/>

    <a href="index.php">Lihat Semua Data</a>

    <br/>
    <h3>Input data baru</h3>
    <form action="input-aksi.php" method="post">
        <table>
            <tr>
                <td>Nama</td>
                <td><input type="text" name="nama"></td>
            </tr>
            <tr>
                <td>Alamat</td>
                <td><input type="text" name="alamat"></td>
            </tr>
            <tr>
                <td>Pekerjaan</td>
                <td><input type="text" name="pekerjaan"></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Simpan"></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

Perhatikan pada syntax di atas. kita membuat form yang mengarah ke input-aksi.php. karena kita menjadikan file *input-aksi.php* sebagai aksi dari form inputan ini. Membuat CRUD Dengan PHP Dan MySQL Input Data . jadi pada file input-aksi.php lah kita menangkap data yang di input dan kemudian mengirimkannya ke database untuk di simpan. oleh karena itu kita buat lagi sebuah file dengan nama *input-aksi.php*.

### input-aksi.php

```

<?php
include 'koneksi.php';
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];

```

```
$pekerjaan = $_POST['pekerjaan'];

mysql_query("INSERT INTO user VALUES('','$nama','$alamat','$pekerjaan')");

header("location:index.php?pesan=input");
?>
```

Perhatikan pada syntax *input-aksi.php* di atas. pertama jangan lupa kita hubungkan dengan databasenya. karena kita di sini memerlukan koneksi database untuk menginput data ke database.

```
include 'koneksi.php';
```

lalu menangkap data yang di kirim dari form dan memasukkannya ke variabel masing-masing.

```
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];
$pekerjaan = $_POST['pekerjaan'];
```

kemudian tinggal menginputnya saja ke database dengan menggunakan fungsi *mysql\_query()*.

```
mysql query("INSERT INTO user VALUES('','$nama','$alamat','$pekerjaan')");
```

dan terakhir setelah menginput data ke database kita alihkan halaman ke halaman index kembali dengan syntax

```
header("location:index.php?pesan=input");
```

coba teman-teeman perhatikan pada syntax hyperlink php di atas. kita mengalihkan halaman kembali ke *index.php* sambil mengirim pesan yang berisi input pada parameter.

```
index.php?pesan=input
```

ini bertujuan untuk membuat pesan bahwa data telah berhasil di input. coba teman-teeman lihat file *index.php* yang terdapat penampil pesan yang sudah saya buat dan saya bilang akan saya jelaskan di *part* ini.

```

<?php
if(isset($_GET['pesan'])){
    $pesan = $_GET['pesan'];
    if($pesan == "input"){
        echo "Data berhasil di input.";
    }else if($pesan == "update"){
        echo "Data berhasil di update.";
    }else if($pesan == "hapus"){
        echo "Data berhasil di hapus.";
    }
}
?>

```

karena pesan yang di tangkap adalah “input” maka akan di tampilkan “*data berhasil di input*”. jika teman-teman belum memahaminya teman-teman bisa meninggalkan komentar di bawah atau bisa melihat tutorial saya sebelumnya tentang form handling di php yang menjelaskan tentang penggunaan method POST dann GET pada PHP dalam pengiriman data.

Oke sekarang waktu nya untuk melihat hasil nya. coba klik pada tombol tambah data baru yang akan mengalihkan halaman ke input.php



[Lihat Semua Data](#)

#### Input data baru

Nama	<input type="text" value="Bob marley"/>
Alamat	<input type="text" value="Jamaika"/>
Pekerjaan	<input type="text" value="Penyanyi"/>
<input type="button" value="Simpan"/>	

menginput data ke database dengan php dan mysql

Nah form input pun jadi. dan coba kita input data baru. saya menginput nama bob marley, alamat jamaika dan pekerjaan nya penyanyi. kemudian kita tes klik simpan.

# Membuat CRUD Dengan PHP Dan MySQL

Menampilkan data dari database

www.malasngoding.com

Data berhasil di input.  
+ Tambah Data Baru



## Data user

No	Nama	Alamat	Pekerjaan	Opsi
1	Andi	Surabaya	web programmer	Edit   Hapus
2	Santoso	Jakarta	Web Designer	Edit   Hapus
3	Samsul	Sumedang	Pegawai	Edit   Hapus
4	Bob marley	Jamaika	Penyanyi	Edit   Hapus



membuat crud dengan php dan mysql input data ke database

Nah dapat teman-teman lihat bahwa data telah berhasil tersimpan. bersamaan dengan memunculkan pesan data berhasil di input.

## Edit & Update Data Dari Database

Pada tutorial membuat crud dengan php dan mysql sebelumnya kita telah membahas tentang cara menampilkan data dari database dan cara menginput data ke database dengan php. sekarang pada sub judul ini kita akan membahas tentang cara mengedit dan mengupdate data pada database dengan PHP.

Untuk membuat edit dan update untuk mengubah data pada database dengan php, caranya gini. teman-teman tangkap dulu id data yang mau di edit. dan kemudian kita tampilkan data menurut id tersebut di dalam form edit. setelah data di ubah pada form edit selanjutnya buat sebuah file php lagi untuk menangani form handling dari form edit tadi. tangkap semua data yang sudah di ubah dan update dengan perintah mysql menurut id. pasti teman-teman yang baru belajar bingung kan ? tetapi jangan khawatir, silahkan teman-teman simak penjelasan berikut ini untuk membuat crud dengan php dan mysql lebih tepatnya membuat edit dan update data untuk mengubah data pada database dengan menggunakan PHP.

coba teman-teeman perhatikan pada file index.php yang menampilkan data. di sana terdapat link edit yang sudah kita buat untuk mengarahkan halaman ke halaman form edit dengan sekaligus mengirimkan id data yang mau di edit.

```
<a href="edit.php?id=<?php echo $data['id']; ?>">Edit</a>
```

Dapat teman-teeman lihat pada syntax pengalihan ke file edit.php di atas. oleh karena itu kita akan membuat sebuah file dengan nama *edit.php* untuk membuat form edit. nah form edit ini lah yg nantinya akan kita tampilkan data yang mau di edit di dalam nya.

### edit.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat CRUD Dengan PHP Dan MySQL - Menampilkan data dari database</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <div class="judul">
        <h1>Membuat CRUD Dengan PHP Dan MySQL</h1>
        <h2>Menampilkan data dari database</h2>
        <h3>www.malasngoding.com</h3>
    </div>

    <br/>

    <a href="index.php">Lihat Semua Data</a>

    <br/>
    <h3>Edit data</h3>

    <?php
    include "koneksi.php";
    $id = $_GET['id'];
    $query_mysql = mysql_query("SELECT * FROM user WHERE id='$id'") or die(mysql_error());
    $nomor = 1;
    while($data = mysql_fetch_array($query_mysql)) {
    ?>
    <form action="update.php" method="post">
        <table>
            <tr>
                <td>Nama</td>
                <td>
                    <input type="hidden" name="id" value="<?php echo $data['id'] ?>">
                    <input type="text" name="nama" value="<?php echo $data['nama'] ?>">
                </td>
            </tr>
        </table>
    </form>

```

```

        </td>
    </tr>
    <tr>
        <td>Alamat</td>
        <td><input type="text" name="alamat" value="<?php
echo $data['alamat'] ?>"></td>
    </tr>
    <tr>
        <td>Pekerjaan</td>
        <td><input type="text" name="pekerjaan" value="<?php
echo $data['pekerjaan'] ?>"></td>
    </tr>
    <tr>
        <td></td>
        <td><input type="submit" value="Simpan"></td>
    </tr>
</table>
</form>
<?php } ?>
</body>
</html>

```

sekarang coba teman-teman klik edit pada data user. dan akan muncul data yang bisa di edit di dalam form.



Lihat Semua Data

Edit data

Nama	<input type="text" value="Bob marley"/>
Alamat	<input type="text" value="Jamaika"/>
Pekerjaan	<input type="text" value="Penyanyi"/>
<input type="button" value="Simpan"/>	



membuat crud dengan php dan mysql edit data

coba perhatikan pada url di atas. url di alihkan sesuai dengan redirect dari edit. dan akan saya jelaskan tentang kegunaan-kegunaan dari syntax-syntax di atas.

```

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

```

```
$query_mysql = mysql_query("SELECT * FROM user WHERE id='$id'") or  
die(mysql_error());  
$nomor = 1;  
while($data = mysql_fetch_array($query_mysql)) {  
?>
```

Coba teman-teeman perhatikan pada syntax di atas. karena kita akan bekerja dengan berhubungan dengan database. oleh sebab itu kita hubungkan dulu php dengan mysql nya.

```
include "koneksi.php";
```

kemudian kita tangkap data id yang di kirimkan melalui url.

```
$id = $_GET['id'];
```

dan kemudian menampilkan data dari database menurut id yang di tangkap ini.

```
$query_mysql = mysql_query("SELECT * FROM user WHERE id='$id'") or  
die(mysql_error());
```

Bisa teman-teeman lihat pada syntax mysql\_query di atas. seperti yang sudah saya jelaskan sebelumnya bahwa mysql\_query adalah fungsi php yang berguna untuk menjalankan perintah mysql. jadi di sini kita menampilkan data user yang ber id sesuai dengan id yang di kirimkan tadi. pada contoh di atas data yang di tampilkan adalah data user yang ber id 64. dan sehingga di tampilkan lah data user yang ber id 64. yaitu id si *bob marley*.

```
<form action="update.php" method="post">  
    <table>  
        <tr>  
            <td>Nama</td>  
            <td>  
                <input type="hidden" name="id" value="<?php echo  
$data['id'] ?>">  
                <input type="text" name="nama" value="<?php echo  
$data['nama'] ?>">  
            </td>  
        </tr>  
        <tr>  
            <td>Alamat</td>  
            <td><input type="text" name="alamat" value="<?php echo  
$data['alamat'] ?>"></td>  
        </tr>
```

```

<tr>
    <td>Pekerjaan</td>
    <td><input type="text" name="pekerjaan" value=<?php echo
$data['pekerjaan'] ?>"></td>
</tr>
<tr>
    <td></td>
    <td><input type="submit" value="Simpan"></td>
</tr>
</table>
</form>
<?php } ?>
```

Method pengiriman data pada form ini kita menggunakan post dan action nya ke file update.php

```
<form action="update.php" method="post">
```

Kita juga mengirimkan data id yang sedang di edit ini. agar nanti di file update.php jadi ketauan data yang ber id mana yg akan di ubah.

```
<input type="hidden" name="id" value="<?php echo $data['id'] ?>">
```

Jadi pada saat data submit halaman akan di alihkan ke update.php. oleh sebab itu kita buat lagi sebuah file update.php untuk memproses form ini.

**update.php**

```

<?php

include 'koneksi.php';
$id = $_POST['id'];
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];
$pekerjaan = $_POST['pekerjaan'];

mysql_query("UPDATE user SET nama='$nama', alamat='$alamat',
pekerjaan='$pekerjaan' WHERE id='$id'");

header("location:index.php?pesan=update");
?>
```

Penjelasan file update.php, kita tangkap dulu data yang di kirim dari form menggunakan method POST sesusi dengan method form yang telah kita tetapkan sebelumnya.

```
$id = $_POST['id'];
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];
$pekerjaan = $_POST['pekerjaan'];
```

Jika teman-teman belum memahami cara pemrosesan sebuah form dengan php (GET dan POST), teman-teman bisa melihat tutorial sebelumnya untuk memahami tentang penggunaan POST dan GET pada PHP.

Selanjutnya kita akan mengupdate data dengan perintah mysql juga. menggunakan bantuan *mysql\_query()* untuk menjalankan perintah mysql untuk mengupdate data. jadi di sini kita akan mengupdate data yang ber id sesuai dengan id yang di kirim.

```
mysql_query("UPDATE user SET nama='$nama', alamat='$alamat',
pekerjaan='$pekerjaan' WHERE id='$id'");
```

Terakhir kita alihkan halaman nya ke index.php kembali sekaligus dengan mengirimkan pesan *update*. agar pada data user (index.php) bisa kita tampilkan juga pesan notifikasi data berhasil di update.

```
header("location:index.php?pesan=update");
```

oke . jadi kesimpulannya

edit.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat CRUD Dengan PHP Dan MySQL - Menampilkan data dari database</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <div class="judul">
        <h1>Membuat CRUD Dengan PHP Dan MySQL</h1>
        <h2>Menampilkan data dari database</h2>
        <h3>www.malasngoding.com</h3>
    </div>
    <br/>
    <a href="index.php">Lihat Semua Data</a>
```

```

<br/>
<h3>Edit data</h3>

<?php
include "koneksi.php";
$id = $_GET['id'];
$query_mysql = mysql_query("SELECT * FROM user WHERE id='$id'") or
die(mysql_error());
$nomor = 1;
while($data = mysql_fetch_array($query_mysql)) {
?>
<form action="update.php" method="post">
    <table>
        <tr>
            <td>Nama</td>
            <td>
                <input type="hidden" name="id" value="<?php
echo $data['id'] ?>">
                <input type="text" name="nama" value="<?php
echo $data['nama'] ?>">
            </td>
        </tr>
        <tr>
            <td>Alamat</td>
            <td><input type="text" name="alamat" value="<?php
echo $data['alamat'] ?>"></td>
        </tr>
        <tr>
            <td>Pekerjaan</td>
            <td><input type="text" name="pekerjaan" value="<?php
echo $data['pekerjaan'] ?>"></td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" value="Simpan"></td>
        </tr>
    </table>
</form>
<?php } ?>
</body>
</html>

```

## update.php

```

<?php

include 'koneksi.php';
$id = $_POST['id'];
$nama = $_POST['nama'];
$alamat = $_POST['alamat'];
$pekerjaan = $_POST['pekerjaan'];

mysql_query("UPDATE user SET nama='$nama', alamat='$alamat',

```

```
pekerjaan='$pekerjaan' WHERE id='$id'");  
header("location:index.php?pesan=update");  
?>
```

Sekarang kita coba merubah data dan klik simpan.

The screenshot shows a web browser window with the URL `localhost/crud_malasngoding/edit.php?id=64`. The page title is "Membuat CRUD Deng Menampilkan dat" and the website address is `www.malasng`. Below the title, there is a link "Lihat Semua Data". The main content is titled "Edit data". It contains three input fields: "Nama" with value "Bob marley", "Alamat" with value "Jakarta", and "Pekerjaan" with value "Musisi". A blue arrow points from the text "update data pada database dengan php dan mysql" to the "Simpan" button. The "Simpan" button is located at the bottom right of the form.

Nama	Bob marley
Alamat	Jakarta
Pekerjaan	Musisi

Simpan

update data pada database dengan php dan mysql

Bisa teman-teeman lihat pada form edit di atas. yang semula nya alamat si bob marley ini adalah jamaika saya ubah menjadi jakarta. dan pekerjaan yang semula penyanyi saya ubah menjadi musisi. sekarang klik simpan.

Data berhasil di update.

+ Tambah Data Baru

No	Nama	Alamat	Pekerjaan	Opsi
1	Andi	Surabaya	web programmer	Edit   Hapus
2	Santoso	Jakarta	Web Designer	Edit   Hapus
3	Samsul	Sumedang	Pegawai	Edit   Hapus
4	Bob marley	Jakarta	Musisi	Edit   Hapus

membuat crud dengan php dan mysql update data

Dan data pun berhasil kita update. data yang semula nya berisi alamat nya jamaika dan pekerjaannya penyanyi sekarang menjadi alamat nya jakarta dan pekerjaan nya musisi. jika teman-teman ingin mengubah data namanya juga bisa. tetapi di sini saya tidak mengubah namanya untuk mencoba apakah semua pengubahan sudah benar atau masih terdapat error. dan hasilnya semua nya berjalan dengan yang di inginkan.

Oke sekian dulu tutorial kita kali ini tentang cara membuat crud dengan PHP dan MySQL edit data. kita telah belajar tentang cara membuat form edit yang berisi data user yang akan di edit dan juga belajar membuat penanganan form edit ini yang berfungsi untuk mengubah data user sesuai dengan yang di inginkan. selanjutnya akan kita buat tentang cara hapus data dari database dengan PHP.

## Menghapus Data Dari Database

Setelah sebelumnya kita belajar tentang cara membuat dengan php dan mysql. yaitu cara menampilkan data dari database, cara menginput data ke database dengan php, dan cara mengedit data pada database. dan sekarang kita akan belajar tentang cara menghapus data dari database dengan php dan database mysql.

Untuk membuat hapus data dari database logikanya juga sama seperti yang telah kita pelajari sebelumnya tentang cara menginput data ke database dan cara mengupdate database dengan php. cuma penggunaannya saja yang agak berbeda. teman-teman juga di tekankan untuk memahami tentang penanganan form pada PHP yaitu pengolahan data dengan metode GET dan metode POST Pada PHP.

Baiklah langsung kita masuk ke topik pokok pembahasan kita di tutorial ini yaitu cara membuat hapus data pada PHP. untuk membuat hapus data pada PHP yang harus teman-teman buat pertama kali adalah sebuah file hapus.php. karena pada index.php(data user) kita membuat link hapus dengan mengarahkannya pada file hapus.php.

```
<a href="hapus.php?id=<?php echo $data['id']; ?>">Hapus</a>  
hapus.php
```

```
<?php  
include 'koneksi.php';  
$id = $_GET['id'];  
mysql_query("DELETE FROM user WHERE id='$id'")or die(mysql_error());  
  
header("location:index.php?pesan=hapus");  
?>
```

Saya jelaskan sedikit tentang syntax-syntax yang ada di file hapus.php ini. karena kita masih memerlukan koneksi ke database kita hubungkan dulu ke file koneksi.php

```
include 'koneksi.php';
```

Kemudian kita tangkap data id dan memasukkan nya ke variabel. id di sini adalah id data yang ingin di hapus.

```
$id = $_GET['id'];
```

Kemudian kita menghapus data pada table user yang ber id sesuai dengan id yang di kirikan tadi.

```
mysql_query("DELETE FROM user WHERE id='$id'")or die(mysql_error());
```

Kita menghapus data dengan perintah mysql *DELETE FROM user WHERE id='id'*. dan terakhir kita alihkan halamannya kembali ke index.php setelah menghapus data.

```
header("location:index.php?pesan=hapus");
```

kita coba jalankan dengan menghapus data si bob marley tadi.

Data berhasil di hapus.   
+ Tambah Data Baru

#### Data user

No	Nama	Alamat	Pekerjaan	Opsi
1	Andi	Surabaya	web programmer	Edit   Hapus
2	Santoso	Jakarta	Web Designer	Edit   Hapus
3	Samsul	Sumedang	Pegawai	Edit   Hapus



membuat crud dengan php dan mysql hapus data

dan data pun berhasil di hapus. contoh aplikasi crud dengan php dan mysql ini hanyalah contoh sederhana yg saya buat. saya membuat nya sesederhana mungkin untuk memudahkan teman-teman yang baru belajar CRUD di PHP lebih mudah memahaminya.

Dari tutorial crud php sebelumnya kita telah berhasil membuat penampilan data dari database, menginput data ke database, mengupdate serta menghapus data pada database. sudah lengkap berarti kita telah berhasil membuat pengelolaan database sederhana dengan menggunakan PHP dan databasenya MySQL.

# Membuat Form Pencarian dengan PHP dan MySQL

Pada tutorial ini kita akan membuat sebuah form pencarian. untuk studi kasusnya, di sini kita membuat pencarian data mahasiswa dengan php. pertama kita harus punya data dulu di database. untuk contoh di tutorial ini, silahkan teman-teman buat sebuah database dengan nama “*mahasiswa*”. dan buat sebuah tabel dengan nama “*mhs*” yang berisi *id* dan *nama*. jika mau cepat silahkan teman import file sql berikut ke database mahasiswa.

```
-- phpMyAdmin SQL Dump
-- version 4.5.1
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: 04 Okt 2017 pada 07.42
-- Versi Server: 10.1.16-MariaDB
-- PHP Version: 5.5.38

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `mahasiswa`
--


-----
-- Struktur dari tabel `mhs`


CREATE TABLE `mhs` (
  `id` int(11) NOT NULL,
  `nama` varchar(35) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data untuk tabel `mhs`


INSERT INTO `mhs` (`id`, `nama`) VALUES
(1, 'Jamaludin'),
(2, 'Sanusi'),
(3, 'Muhammad Haikal'),
(4, 'Roki sinamuna'),
(5, 'Diki Alfarabi Hadi'),
```

```
(6, 'Samsul Bahri'),
(7, 'Khairul Bahri'),
(8, 'Samsul Arifin'),
(9, 'Sumantini'),
(10, 'Aisyah Nur Hikmah');

-- 
-- Indexes for dumped tables
-- 

-- 
-- Indexes for table `mhs`
-- 
ALTER TABLE `mhs`
  ADD PRIMARY KEY (`id`);

-- 
-- AUTO_INCREMENT for dumped tables
-- 

-- 
-- AUTO_INCREMENT for table `mhs`
-- 
ALTER TABLE `mhs`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

pastikan sudah di import dengan benar ya.

Menampilkan baris 0 - 9 (total 10, Pencarian dilakukan dalam 0.0007 detik.)

```
SELECT * FROM `mhs`
```

[ Edit dikotak ] [ Ubah ] [ Jelaskan SQL ] [ Create PHP ]

Tampilan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Urut berdasarkan kunci: Tidak ada

	+ Opsi		id	nama
<input type="checkbox"/>	Ubah  Salin  Hapus	1	Jamaludin	
<input type="checkbox"/>	Ubah  Salin  Hapus	2	Sanusi	
<input type="checkbox"/>	Ubah  Salin  Hapus	3	Muhammad Haikal	
<input type="checkbox"/>	Ubah  Salin  Hapus	4	Roki sinamuna	
<input type="checkbox"/>	Ubah  Salin  Hapus	5	Diki Alfarabi Hadi	
<input type="checkbox"/>	Ubah  Salin  Hapus	6	Samsul Bahri	
<input type="checkbox"/>	Ubah  Salin  Hapus	7	Khairul Bahri	
<input type="checkbox"/>	Ubah  Salin  Hapus	8	Samsul Arifin	
<input type="checkbox"/>	Ubah  Salin  Hapus	9	Sumantini	
<input type="checkbox"/>	Ubah  Salin  Hapus	10	Aisyah Nur Hikmah	

Beberapa data mahasiswa yang telah berhasil di input/import

Setelah selesai dengan databasenya, selanjutnya buat 2 buah file php, yaitu *config.php* dan *index.php*

### config.php

```
<?php
// www.malasngoding.com
// koneksi mysql ke php
mysql_connect("localhost","root","");
mysql_select_db("mahasiswa");
?>
```

Pada file config.php ini kita membuat koneksi php dan mysql. Silahkan baca pada sub judul sebelumnya tentang cara membuat koneksi database php dan mysql jika teman-teman belum membacanya.

### index.php

```
<?php
include 'config.php';
?>
```

```

<h3>Form Pencarian Dengan PHP - WWW.MALASNGODING.COM</h3>

<form action="index.php" method="get">
    <label>Cari :</label>
    <input type="text" name="cari">
    <input type="submit" value="Cari">
</form>

<?php
if(isset($_GET['cari'])){
    $cari = $_GET['cari'];
    echo "<b>Hasil pencarian : ".$cari."</b>";
}
?>

<table border="1">
    <tr>
        <th>No</th>
        <th>Nama</th>
    </tr>
    <?php
    if(isset($_GET['cari'])){
        $cari = $_GET['cari'];
        $data = mysql_query("select * from mhs where nama like
        '%".$cari."%'");
    }else{
        $data = mysql_query("select * from mhs");
    }
    $no = 1;
    while($d = mysql_fetch_array($data)){
    ?>
    <tr>
        <td><?php echo $no++; ?></td>
        <td><?php echo $d['nama']; ?></td>
    </tr>
    <?php } ?>
</table>

```

Sekarang coba jalankan pada browser.

[localhost/pencarian\\_php/index.php](http://localhost/pencarian_php/index.php)

← → C localhost/pencarian\_php/index.php

## Form Pencarian Dengan PHP - WWW.MALASNGODING.COM

Cari :  Cari

No	Nama
1	Jamaludin
2	Sanusi
3	Muhammad Haikal
4	Roki sinamuna
5	Diki Alfarabi Hadi
6	Samsul Bahri
7	Khairul Bahri
8	Samsul Arifin
9	Sumantini
10	Aisyah Nur Hikmah



pencarian dengan php dan mysql

dan kita akan mencoba mencari nama “samsul” pada form pencarian.

dan hasilnya

← → C localhost/pencarian\_php/index.php?cari=samsul

## Form Pencarian Dengan PHP - WWW.MALASNGODING.COM

Cari :  Cari

Hasil pencarian : samsul

No	Nama
1	Samsul Bahri
2	Samsul Arifin

Hasil pencarian data mahasiswa

dan selesai. nama mahasiswa yang mirip dengan “samsul” pun berhasil di tampilkan.

Jika teman-teman bingung, silahkan perhatikan penjelasan di bawah ini tentang penjelasan tutorial cara membuat pencarian dengan php dan mysql.

## Penjelasan

Perhatikan pada form.

```
<h3>Form Pencarian Dengan PHP - WWW.MALASNGODING.COM</h3>

<form action="index.php" method="get">
    <label>Cari :</label>
    <input type="text" name="cari">
    <input type="submit" value="Cari">
</form>
```

Kita memberikan action ke file index.php itu sendiri. dengan method get. sehingga data yang dikirim dari form akan di kirim melalui url.

dan perhatikan juga pada

```
<?php
if(isset($_GET['cari'])){
    $cari = $_GET['cari'];
    $data = mysql_query("select * from mhs where nama like '%".$cari."%'");

} else{
    $data = mysql_query("select * from mhs");
}
```

Disini kita memeriksa apakah ada data yang dikirim dengan menggunakan method GET. jika ada maka query mysql yang di jalankan adalah query berikut.

```
$data = mysql_query("select * from mhs where nama like '%".$cari."%'");
```

query di atas adalah query mysql untuk mencari data.

jika tidak ada, maka query mysql yang di jalankan adalah query berikut, karena jika tidak ada pencarian kita akan menampilkan semua data mahasiswa.

```
$data = mysql_query("select * from mhs");
```

Dan selesai.

# Membuat Upload File Dengan PHP Dan MySQL

Upload file adalah salah satu yang terpenting di dalam membuat aplikasi. khususnya aplikasi berbasis web. karena terkadang kita di haruskan membuat upload file, bisa itu fitur upload file laporan, atau mungkin upload file gambar. seperti pada facebook contoh nya. di facebook terdapat fitur untuk mengupload gambar untuk di jadikan foto profil. dan contoh aplikasi lain yang harus ada fitur upload file nya adalah aplikasi manajemen arsip. contoh nya manajemen arsip kantor. di mana data-data pdf atau document lainnya bisa di upload untuk di arsipkan.

Pada tutorial kali ini saya akan share tutorial cara Membuat Upload File Dengan PHP Dan MySQL. dan menyimpan data gambar nya ke database. di tutorial ini kita akan belajar tentang cara membuat upload file dengan php. dan kemudian nama dari file tersebut akan kita simpan pada database.

## Persiapan Membuat Upload File

Untuk persiapan nya. apa-apa saja yang harus teman-teman buat adalah sebagai berikut. buat lah:

- index.php
- aksi.php
- koneksi.php
- folder bernama file

The screenshot shows a Windows File Explorer window with the following directory path in the address bar: Computer > Local Disk (C:) >xampp >htdocs > malasngoding > tutorial > fileupload >. Below the address bar, there are buttons for 'Search in library' and 'Share with'. The main area displays a list of files and folders:

Name	Date modified	Type	Size
file	2/27/2016 12:08 PM	File folder	
aksi	2/27/2016 12:09 PM	PHP File	2 KB
index	2/27/2016 11:53 AM	PHP File	1 KB
koneksi	2/27/2016 11:51 AM	PHP File	1 KB

membuat upload file dengan php dan mysql

Akan saya jelaskan sedikit kegunaan-kegunaan dari file-file yang telah di buat di atas. pada file index.php kita akan membuat form upload file gambar nya. pada file aksi.php kita akan membuat pemroses dari form upload file. jadi file yang di upload dari index.php akan di proses di file aksi.php. kemudian pada file koneksi.php kita akan membuat koneksi database php dan mysql. baca tutorial sebelumnya tentang cara membuat koneksi php dengan database mysql. lalu yang terakhir folder yang bernama file akan kita gunakan sebagai folder yang menyimpan file gambar yang di upload. jadi file gambar yang di upload akan tersimpan ke folder *file* ini.

## Membuat Upload File

Langsung saja kita mulai. buatlah koneksi database nya dulu.

koneksi.php

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

Silahkan teman-teman sesuaikan password dan yg lainnya. di sini kita menggunakan database dengan nama “malasngoding”. jadi silahkan teman-teman buat database dengan nama *malasngoding*. lalu buat table dengan nama upload. atau biar lebih cepat nya silahkan teman-teman import saja file sql berikut ini.

```
-- phpMyAdmin SQL Dump  
-- version 3.5.2.2  
-- http://www.phpmyadmin.net  
--  
-- Host: 127.0.0.1  
-- Generation Time: Feb 27, 2016 at 06:34 AM  
-- Server version: 5.5.27  
-- PHP Version: 5.4.7  
  
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";  
SET time_zone = "+00:00";  
  
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;  
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;  
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;  
/*!40101 SET NAMES utf8 */;  
  
--  
-- Database: `malasngoding`  
--
```

```

-- -----
-- 
-- Table structure for table `upload`
-- 

CREATE TABLE IF NOT EXISTS `upload` (
  `id_file` int(11) NOT NULL AUTO_INCREMENT,
  `nama_file` varchar(100) NOT NULL,
  PRIMARY KEY (`id_file`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

Selanjutnya ketikkan perintah berikut pada file index.php. karena kita akan membuat form upload file di file index.php ini.

### index.php

```

<!DOCTYPE html>
<html>
<head>
    <title>Membuat Upload File Dengan PHP Dan MySQL | www.malasngoding.com</title>
</head>
<body>
    <h1>Membuat Upload File Dengan PHP Dan MySQL <br/> www.malasngoding.com</h1>
    <form action="aksi.php" method="post" enctype="multipart/form-data">
        <input type="file" name="file">
        <input type="submit" name="upload" value="Upload">
    </form>
</body>
</html>
```

perhatikan pada element form di atas.

```
<form action="aksi.php" method="post" enctype="multipart/form-data">
```

Terdapat atribut *enctype* yang berfungsi untuk memberikan intruksi pada php bahwa form ini berupa form untuk mengupload file.

```
enctype="multipart/form-data"
```

Jadi gunakan atribut enctype ini untuk membuat upload file.

lalu kemudian di file aksi.php nya.

### aksi.php

```
<!DOCTYPE html>
<html>
    <head>
        <title>Membuat Upload File Dengan PHP Dan MySQL | www.malasngoding.com</title>
    </head>
    <body>
        <h1>Membuat Upload File Dengan PHP Dan MySQL <br/> www.malasngoding.com</h1>
        <?php
            include 'koneksi.php';
            if($_POST['upload']){
                $ekstensi_diperbolehkan = array('png','jpg');
                $nama = $_FILES['file']['name'];
                $x = explode('.',$nama);
                $ekstensi = strtolower(end($x));
                $ukuran = $_FILES['file']['size'];
                $file_tmp = $_FILES['file']['tmp_name'];

                if(in_array($ekstensi, $ekstensi_diperbolehkan) === true){
                    if($ukuran < 1044070){
                        move_uploaded_file($file_tmp, 'file/'.$nama);
                        $query = mysql_query("INSERT INTO upload VALUES (NULL, '$nama')");
                        if($query){
                            echo 'FILE BERHASIL DI UPLOAD';
                        }else{
                            echo 'GAGAL MENGUPLOAD GAMBAR';
                        }
                    }else{
                        echo 'UKURAN FILE TERLALU BESAR';
                    }
                }else{
                    echo 'EKSTENSI FILE YANG DI UPLOAD TIDAK DI PERBOLEHKAN';
                }
            }
        ?>

        <br/>
        <br/>
        <a href="index.php">Upload Lagi</a>
        <br/>
        <br/>

        <table>
            <?php
                $data = mysql_query("select * from upload");
                while($d = mysql_fetch_array($data)){

```

```

?>
<tr>
    <td>
        
    </td>
</tr>
<?php } ?>
</table>
</body>
</html>

```

saya jelaskan sedikit proses upload nya. perhatikan pada file aksi.php di atas.

```

<?php
include 'koneksi.php';
if($_POST['upload']){
    $ekstensi_diperbolehkan = array('png', 'jpg');
    $nama = $_FILES['file']['name'];
    $x = explode('.', $nama);
    $ekstensi = strtolower(end($x));
    $ukuran = $_FILES['file']['size'];
    $file_tmp = $_FILES['file']['tmp_name'];
    if(in_array($ekstensi, $ekstensi_diperbolehkan) === true){
        if($ukuran < 1044070){
            move_uploaded_file($file_tmp, 'file/'.$nama);
            $query = mysql_query("INSERT INTO upload VALUES(NULL,
'$nama')");
            if($query){
                echo 'FILE BERHASIL DI UPLOAD';
            }else{
                echo 'GAGAL MENGUPLOAD GAMBAR';
            }
        }else{
            echo 'UKURAN FILE TERLALU BESAR';
        }
    }else{
        echo 'EKSTENSI FILE YANG DI UPLOAD TIDAK DI PERBOLEHKAN';
    }
}
?>

```

Pertama karena kita menggunakan database. maka hubungkan dulu dengan file koneksi.php. agar php terhubung dengan database mysql.

```
include 'koneksi.php';
```

dan memasukkan detail dari data yang di upload ke masing-masing variabel

```
$ekstensi_diperbolehkan = array('png', 'jpg');
```

```
$nama = $_FILES['file']['name'];
$x = explode('.', $nama);
$ekstensi = strtolower(end($x));
$ukuran = $_FILES['file']['size'];
$file_tmp = $_FILES['file']['tmp_name'];
```

nah, perintah di atas adalah perintah untuk mendapatkan data file yang di upload. contohnya seperti misalnya:

- `$_FILES['file']['name']` = berguna untuk mendapatkan nama file yang di upload.
- `$_FILES['file']['size']` = mendapatkan ukuran file yang di upload.
- `$_FILES['file']['tmp_name']` = untuk mendapatkan temporary file yang di upload (`tmp`).

Pada variabel `$ekstensi_diperbolehkan` saya memberikan nama-nama ekstensi file yang di perbolehkan untuk di upload. ini berguna untuk keamanan. jadi user nantinya tidak bisa mengupload sembarang file karena alasan keamanan. jadi di sini kita hanya memperbolehkan untuk mengupload file gambar saja. yang berekstensi png dan jpg. teman-teman bisa menyesuaikan nya lagi nanti. karena di sini saya hanya membuat contoh dan penjelasannya saja tentang cara membuat upload file gambar dengan php dan mysql.

```
$ekstensi_diperbolehkan = array('png', 'jpg');
```

## Proses Upload File Gambar

Dan yang paling terpenting di sini adalah di bagian ini.

```
if(in_array($ekstensi, $ekstensi_diperbolehkan) === true){
    if($ukuran < 1044070){
        move_uploaded_file($file_tmp, 'file/'.$nama);
        $query = mysql_query("INSERT INTO upload VALUES(NULL, '$nama')");
        if($query){
            echo 'FILE BERHASIL DI UPLOAD';
        }else{
            echo 'GAGAL MENGUPLOAD GAMBAR';
        }
    }else{
        echo 'UKURAN FILE TERLALU BESAR';
    }
}else{
    echo 'EKSTENSI FILE YANG DI UPLOAD TIDAK DI PERBOLEHKAN';
}
```

Di sini kita memeriksa ekstensi file yang di upload dengan mengecek apakah ekstensi file yang di upload sesuai dengan nama ekstensi yang telah kita tetapkan di variabel array \$ekstensi\_diperbolehkan tadi. jika tidak maka tidak akan di upload dan di tampilkan pesan bahwa ekstensi file yang di upload tidak di perbolehkan.

Lalu memeriksa ukuran file yang di upload. kita menetapkan ukuran maksimalnya sebesar 1 mb. jadi di sini percabangan if else PHP juga sangat berpengaruh ya teman-teman.

```
if($ukuran < 1044070) {
```

Jika file yang di upload lebih besar dari 1 mb maka akan di tampilkan pesan "ukuran file terlalu besar". jika ukuran file nya sesuai maka proses upload akan di lanjutkan lagi.

```
move_uploaded_file($file_tmp, 'file/'.$nama);
$query = mysql_query("INSERT INTO upload VALUES(NULL, '$nama')");
if($query){
    echo 'FILE BERHASIL DI UPLOAD';
} else{
    echo 'GAGAL MENGUPLOAD GAMBAR';
}
```

Di sini lah inti upload file dengan php dan mysql nya. fungsi atau function *move\_uploaded\_file()* berfungsi untuk mengupload. atau dengan kata lain dengan fungsi ini lah yang bertugas untuk memindahkan file yang di upload tadi dari temporary file menuju ke folder file. dan memberikan nama file nya sesuai yang kita tentukan.

```
move_uploaded_file($file_tmp, 'file/'.$nama);
```

Jadi di sini teman-teman bisa menentukan tujuan file upload nya kemana. maksudnya di simpan ke mana file yang di upload. karena di sini kita telah menyediakan folder "file" untuk menyimpan gambar nya maka kita tujuhan ke folder file.

Dan terakhir kita insert data nama file yang di upload ke database. tutorial cara input/insert data ke database sudah saya bahas sebelumnya.

```
$query = mysql_query("INSERT INTO upload VALUES(NULL, '$nama')");
```

oke akan kita jalankan langsung. Untuk melihat hasilnya.



## **Membuat Upload File Dengan PHP Dan MySQL**

### **www.malasngoding.com**

dp malas ngoding.PNG

membuat upload file gambar dengan php dan mysql

lalu klik upload.

## Membuat Upload File Dengan PHP Dan MySQL

[www.malasngoding.com](http://www.malasngoding.com)

FILE BERHASIL DI UPLOAD

[Upload Lagi](#)



Gambar berhasil di upload dan berhasil muncul

Oke selesai teman-teman. Gambar telah berhasil terupload dan berhasil muncul. Pada database juga telah tersimpan nama file gambar yang diupload ini.

# Membuat Pagination atau Penomoran Halaman Dengan PHP dan MySQL

Pagination atau paging merupakan suatu teknik untuk menampilkan data dalam beberapa halaman yang berbeda. kenapa kita menggunakan paging, itu karena dalam menampilkan banyak data dalam halaman tidaklah efektif, hal ini selain dapat membuat load data dari database semakin besar dan dapat membuat user merasa bosan karena harus scroll halaman yang begitu banyak.

Untuk memecahkan masalah ini, maka teknik paging menjadi salah satu solusi yang paling tepat, karena data yang kita tampilkan dalam beberapa halaman, dan jumlah data yang kita tampilkan juga dibatasi. misalkan kita akan menampilkan 10 data dalam setiap halaman paging.

Langsung dalam contoh kasus Membuat Paging Dengan PHP dan MySQL. hal yang pertama yang harus di persiapkan yaitu database.

berikut contohnya :

	Opsi	id	nama_msj	alamat	kecamatan	latitude	longitude	gambar	keterangan
	[Edit] [Salin] [Hapus]	20	Masjid Raudhatul Munawwarah	Lhok Seutuy	Baktiya	5.135621	97.415085		masjid jami
	[Edit] [Salin] [Hapus]	21	Masjid Baitul Muhaqirin	Alue Dama	Baktiya	5.126099	97.435844		
	[Edit] [Salin] [Hapus]	22	Masjid Raudhatul Jinan	Mns Geudong	Baktiya	5.142938	97.442322		
	[Edit] [Salin] [Hapus]	23	Masjid Baitul Maghfirah	Matang Kumbang	Baktiya				
	[Edit] [Salin] [Hapus]	24	Masjid Baitul Abidin	Geulumpang Payong	Baktiya				
	[Edit] [Salin] [Hapus]	25	Masjid Baitul Ghafur	Babussalam	Baktiya				
	[Edit] [Salin] [Hapus]	26	Masjid Baitul Muttaqin	Geulumpang Samlakoe	Baktiya				
	[Edit] [Salin] [Hapus]	27	Masjid Baitul Ibad	Buket Mon Sukon	Baktiya	5.110101	97.353170		
	[Edit] [Salin] [Hapus]	28	Masjid Baitul Mabrur	Matang Lawang	Baktiya	5.110105	97.353174		
	[Edit] [Salin] [Hapus]	29	Masjid Jadid	Krueng Lingka Timu	Baktiya	5.109419	97.351791		
	[Edit] [Salin] [Hapus]	30	Masjid Al-Munawwarah	Keude Alue le Puteh	Baktiya				
	[Edit] [Salin] [Hapus]	31	Masjid Al-Istiqamah	Alue Anoe Barat	Baktiya	5.143074	97.402864		
	[Edit] [Salin] [Hapus]	32	Masjid Baitul Akram	Alue Buya	Baktiya				
	[Edit] [Salin] [Hapus]	33	Masjid Baiturrahman	Keude Alue le Puteh	Baktiya	5.131343	97.405586		
	[Edit] [Salin] [Hapus]	34	Masjid HIDAYATULLAH	COT LABA BUAH	Baktiya Barat				
	[Edit] [Salin] [Hapus]	35	Masjid TAQARRUB	MATANG PANYANG	Baktiya Barat				
	[Edit] [Salin] [Hapus]	36	Masjid Baitul Mukarramah	Matang Raya	Baktiya Barat				

## Membuat Paging Dengan PHP dan MySQL

Dari gambar di atas, dapat dilihat bahwa nama database yang saya contohkan tutorial dengan table *tb\_masjid*. untuk database, table dan isi dalam table itu tergantung bagaimana keinginan tidak harus dicontoh disisni.

langkah yang kedua, membuat file koneksi.php

### koneksi.php

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

File koneksi ini fungsinya untuk mengkoneksikan aplikasi yang kita buat dengan database mysql. Untuk syntax konesinya dapat dilihat di baris syntax di atas.

Langkah terakhir menampilkan data dan membuat paging untuk halaman yang kita tampilkan data tersebut. Buat sebuah file lagi dengan nama paging.php.

### paging.php

```
<?php  
include '../config.php';  
$halaman = 10; //batasan halaman  
$page = isset($_GET['halaman']) ? (int)$_GET["halaman"] : 1;  
$mulai = ($page > 1) ? ($page * $halaman) - $halaman : 0;  
$query = mysql_query("select * from tb_masjid LIMIT $mulai, $halaman");  
$sql = mysql_query("select * from tb_masjid");  
$total = mysql_num_rows($sql);  
$pages = ceil($total / $halaman);  
for ($i=1; $i <= $pages ; $i++) { ?>  
    <a href="?halaman=<?php echo $i; ?>"><?php echo $i; ?></a>  
  
<?php  
}  
?>
```

### Penjelasan

```
$mulai = ($page > 1) ? ($page * $halaman) - $halaman : 0;
```

Syntax ini merupakan perhitungan yang akan kita lakukan untuk halaman.

```
$query = mysql_query("select * from tb_masjid LIMIT $mulai, $halaman");
```

Syntax ini digunakan untuk menampilkan data database dengan membatasi perpage nya.

```
$pages = ceil($total/$halaman);
```

Syntax ini fungsinya akan membulatkan hasil dari total jumlah page dan berikut tampilan dari syntax yang ditas :



## Membuat Paging Dengan PHP dan MySQL

Selanjutnya kita akan mengimplementasikan dalam contoh kasus menampilkan data masjid sebanyak 45 masjid. dan kita memisahkan data yang akan tampil 10 data per page. berikut syntaxnya :

```
<?php
include '../config.php';
?>



| No                    | Nama                            |
|-----------------------|---------------------------------|
| <?php echo \$no++; ?> | <?php echo \$data['alamat']; ?> |


```

```

</tr>

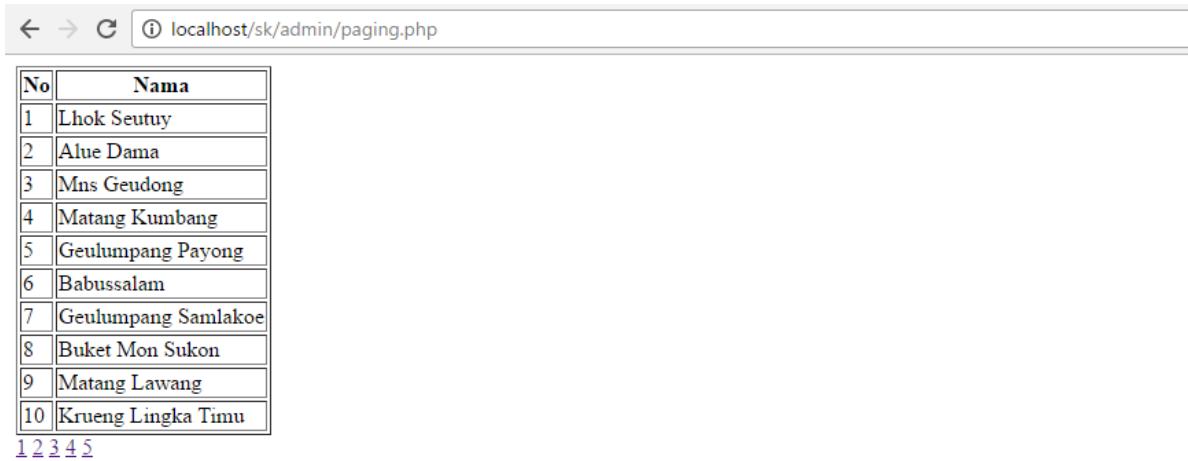
<?php
}
?>

</table>

<div class="">
<?php for ($i=1; $i<=$pages ; $i++){ ?>
<a href="?halaman=<?php echo $i; ?>"><?php echo $i; ?></a>
<?php } ?>
</div>

```

Dan coba jalankan pada browser untuk melihat hasilnya. Akses file paging yang telah teman-teman buat.



The screenshot shows a web browser window with the URL `localhost/sk/admin/paging.php`. The page displays a table with 10 rows, each containing a number from 1 to 10 and a corresponding name. Below the table is a set of page navigation links.

No	Nama
1	Lhok Seutuy
2	Alue Dama
3	Mns Geudong
4	Matang Kumbang
5	Geulumpang Payong
6	Babussalam
7	Geulumpang Samlakoe
8	Buket Mon Sukon
9	Matang Lawang
10	Krueng Lingka Timu

1 2 3 4 5

## Membuat Paging Dengan PHP dan MySQL

A screenshot of a web browser window. The address bar shows the URL: `localhost/sk/admin/paging.php?halaman=5`. Below the address bar is a horizontal navigation bar with icons for back, forward, and refresh. The main content area displays a table with data and a set of page navigation links.

No	Nama
41	Nurul Iman Cot Girek
42	Desa Cot Girek Lama
43	Jln. Beurandang Krueng
44	Gampong Lueng Baro
45	Gampong U Baro

[1](#) [2](#) [3](#) [4](#) [5](#)

### Membuat Paging Dengan PHP dan MySQL

oke, sekian dulu tutorial Membuat Paging Dengan PHP dan MySQL.

# Koneksi Database PHP dan MySQLi

Sebenarnya kita telah membahas di tutorial sebelumnya tentang cara menghubungkan PHP dengan database. tapi menggunakan MySQL. berbeda dengan tutorial di sub judul ini. kali ini kita akan menggunakan MySQLi. di sub judul ini juga akan di bahas tentang kenapa menggunakan MySQLi? dan apa itu MySQLi?.

Menghubungkan PHP dan database adalah suatu keharusan yang dasar dalam pemrograman web. karena dengan menghubungkan PHP dan database, kita bisa mengakses database untuk berinteraksi lebih lanjut. contohnya seperti menginput data ke database, menampilkan data dari database, mengedit data dari database, menghapus data dari database. serta interaksi data ke database lainnya yang akan kita lakukan.

INGAT! di sini kita akan menggunakan MySQLi, bukan lagi MySQL.

## Kenapa Menggunakan MySQLi ?

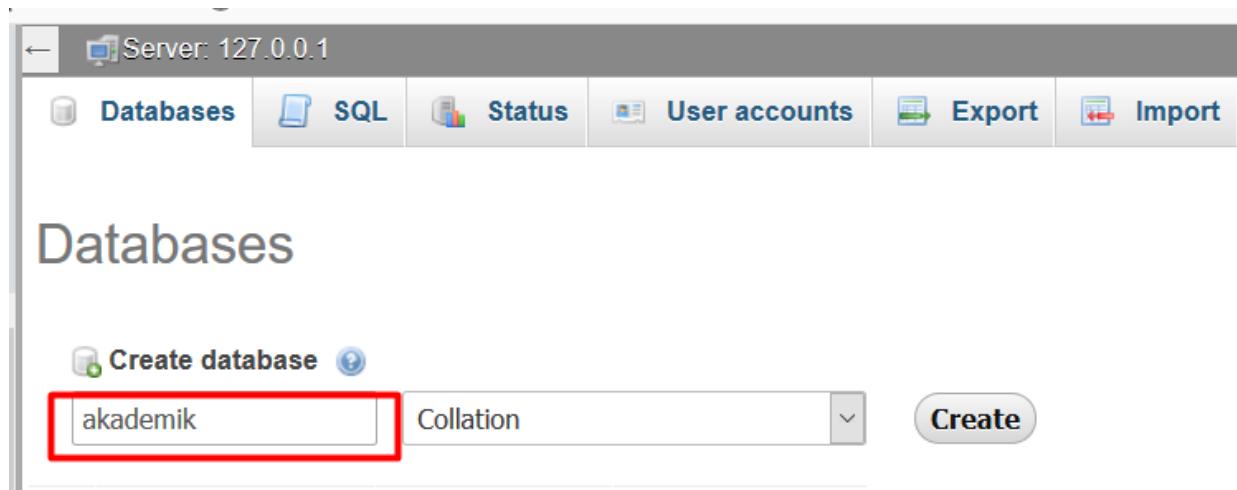
Sebelum membahas cara membuat koneksi database dengan MySQLi, ada baiknya saya jelaskan dulu kenapa kita menggunakan MySQLi, dan apa pengertian MySQLi?

MySQLi adalah sebuah fungsi baru dari MySQL. MySQLi di informasikan lebih aman digunakan. dan lebih cepat dari pada MySQL. MySQLi di ciptakan untuk di jalankan pada versi PHP 5 ke atas (PHP 7). karena lebih stabil. jadi mulai sekarang kita sudah boleh berhenti menggunakan mysql untuk pemrosesan database. dan beralih menggunakan MySQLi.

Jadi intinya di PHP 5 ke atas sudah tidak lagi menggunakan **mysql**, tapi sudah menggunakan **mysqli**.

## Menghubungkan PHP dengan database menggunakan MySQLi

Langsung saja kita mulai cara membuat koneksi database dengan PHP dan MySQLi. di sini kita membutuhkan sebuah database baru untuk contoh. jadi di sini saya membuat database baru dengan nama "*akademik*". teman-teman bebas memberikan nama databasenya. tapi nanti di kondisikan saja.



membuat database baru

Selesai membuat database dengan nama 'akademik'. selanjutnya untuk contoh, buat folder baru dengan nama 'koneksi' dalam folder *htdocs*.

Setelah membuat folder dengan nama *koneksi*, buat file php baru dengan nama *koneksi.php* .

ODE > Local Disk (C) > xampp > htdocs > koneksi				
Name	Date modified	Type	Size	
koneksi	3/8/2018 16:43	PHP File	0 KB	

membuat koneksi database dengan php dan mysqli

dan ketik syntax berikut dalam file *koneksi.php*.

```
<?php  
  
$koneksi = mysqli_connect("localhost","root","","","akademik");  
  
// Check connection  
if (mysqli_connect_errno()) {  
    echo "Koneksi database gagal : " . mysqli_connect_error();  
}  
  
?>
```

oke, saya akan menjelaskan dulu apa saja kegunaan dari syntax di atas. pertama, perhatikan pada syntax :

```
$koneksi = mysqli_connect("localhost","root","","akademik");
```

Pada parameter pertama, kita mengetikkan nama host kita. yaitu *localhost*. dan pada parameter kedua kita mengetikan *username* dari mysql kita, defaultnya adalah *root*. parameter ketiga kita ketikkan *password* mysql kita (jika ada), karena default nya kosong. jika teman-teman menggunakan password pada mysql, silahkan teman-teman ketik pada parameter ke tiga ini. dan terakhir pada parameter ke empat kita mengetik nama database yang ingin kita gunakan. atau database yang ingin kita hubungkan ke php. isikan sesuai dengan nama database yang ingin teman-teman gunakan. karena tadi saya membuat database baru dengan nama *akademik*, maka di sini saya mengetikkan akademik.

lebih jelasnya silahkan perhatikan pada gambar berikut ini.

```
koneksi.php
1 <?php
2 $koneksi = mysqli_connect("localhost","root","","akademik");
3 // Check connection
4 if (mysqli_connect_errno()){
5     echo "Koneksi database gagal : ". mysqli_connect_error();
6 }
7
8 ?>
```

koneksi php dan mysqli

perhatikan juga pada :

```
// Check connection
```

```
if (mysqli_connect_errno()) {  
echo "Koneksi database gagal : " . mysqli_connect_error();  
}
```

di sini kita akan mengecek, apakah koneksi berhasil atau tidak. fungsi *mysqli\_connect\_errno()* berguna untuk memeriksa koneksi apakah berhasil atau tidak, jika gagal maka akan kita jalankan fungsi berikut:

```
echo "Koneksi database gagal : " . mysqli_connect_error();
```

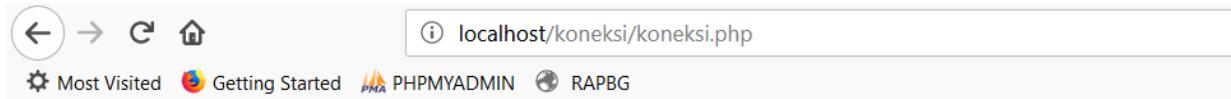
Fungsi *mysqli\_connect\_error()* berguna untuk menampilkan error yang terjadi di koneksi php dan mysqli.

Sekarang akan kita coba jalankan file *koneksi.php*. jika tidak tampil apa-apa, maka koneksi database telah berhasil. jika terdapat error berarti koneksi database php dan mysqli belum sesuai, sehingga harus di sesuaikan lagi.

jalankan pad browser :

(jangan lupa megaktifkan atau menjalankan apache dan mysql terlebih dulu)

dan hasil nya



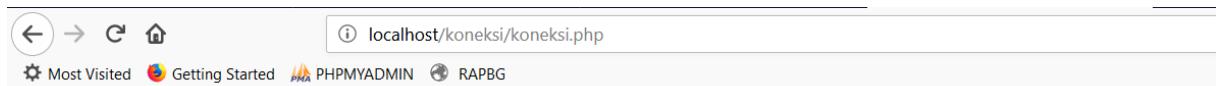
koneksi database php mysqli berhasil

sampai di sini kita telah berhasil membuat koneksi database php dan mysqli.

sekarang coba untuk merubah nama database menjadi "sekolah".

```
$koneksi = mysqli_connect("localhost","root","","sekolah");
```

maka akan muncul pesan error yang memberitahukan bahwa database dengan nama 'sekolah' tidak di temukan.



tes koneksi php dan mysqli

karena database *sekolah* tidak ada di database kita.

Sekian dulu tutorial tentang Membuat Koneksi Database PHP dan MySQLi.

# Membuat CRUD dengan PHP dan MySQLi

Pada tutorial sebelum-sebelumnya, saya telah membuatkan tutorial cara membuat CRUD dengan PHP dan MySQL (tanpa i). tapi pada tutorial kali ini, saya akan membuat tutorial membuat CRUD dengan PHP dan MySQLi (menggunakan i). apa perbedaan dari MySQL dan MySQLi ? sudah sedikit di jelaskan di tutorial sebelumnya, yaitu pada tutorial membuat koneksi database dengan PHP dan MySQLi.

CRUD adalah singkatan dari Create, Read, Update, Delete. ini merupakan istilah dari proses penginputan data ke database (create), menampilkan data dari database (read), mengedit atau mengubah data di database (update), dan menghapus data dari database (delete).

Namun akan sedikit di jelaskan kembali pada teman-teman apa perbedaan dari mysql dan mysqli. dan kenapa kita harus beralih dari mysql ke mysqli ? yuk simak perbedaan dan kenapa kita harus pindah dari mysql ke mysqli.

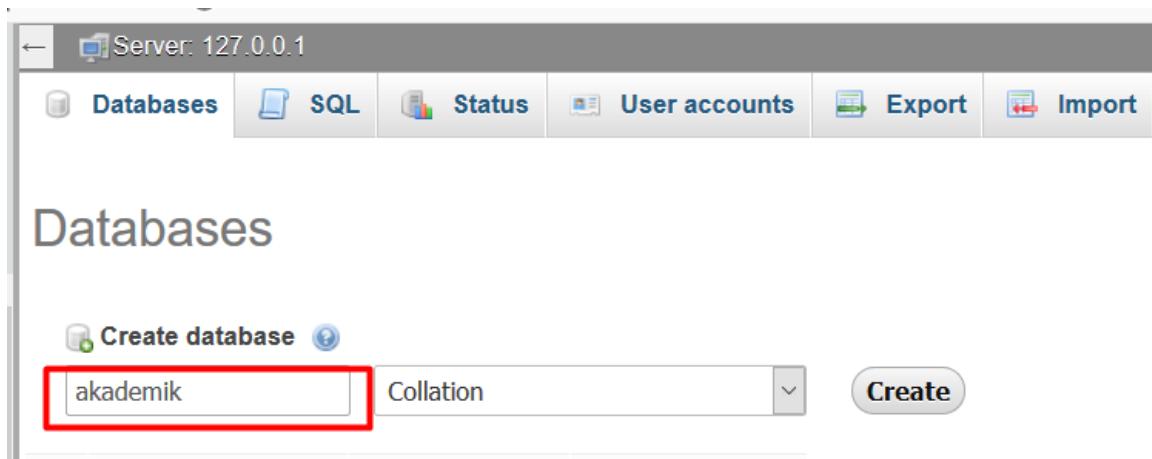
MySQLi bisa dikatakan adalah fitur atau fungsi terbaru dari mysql. kabarnya, mysqli lebih cepat dan lebih aman dari pada mysql. jadi bisa kita simpulkan mysqli adalah pengembangan dari mysql. mysqli dibuat agar membuat proses pemrosesan database lebih cepat dan aman. seperti yang kita ketahui, sekarang PHP sudah sampai pada versi 7. di PHP 7 sudah mengharuskan kita menggunakan mysqli. karena mysql sudah tidak didukung dengan berbagai alasan seperti yang di jelaskan sebelumnya.

Oke kita masuk ke materi tentang tutorial membuat CRUD dengan PHP dan MySQLi. seperti biasa, disini kita akan bermain-main dengan database. oleh sebab itu kita perlu menghubungkan project PHP kita dengan database dengan cara membuat file koneksi.

## Persiapan Database

Saya yakin teman-teman pasti sudah paham cara membuat database. karena di tutorial-tutorial sebelumnya juga kita telah banyak belajar cara membuat database.

Buat database dengan nama “*akademik*”. Kemudian kita buat sebuah table dengan nama “*mahasiswa*”.



membuat database baru

kemudian kolum nya buat id, nama, nim dan alamat.

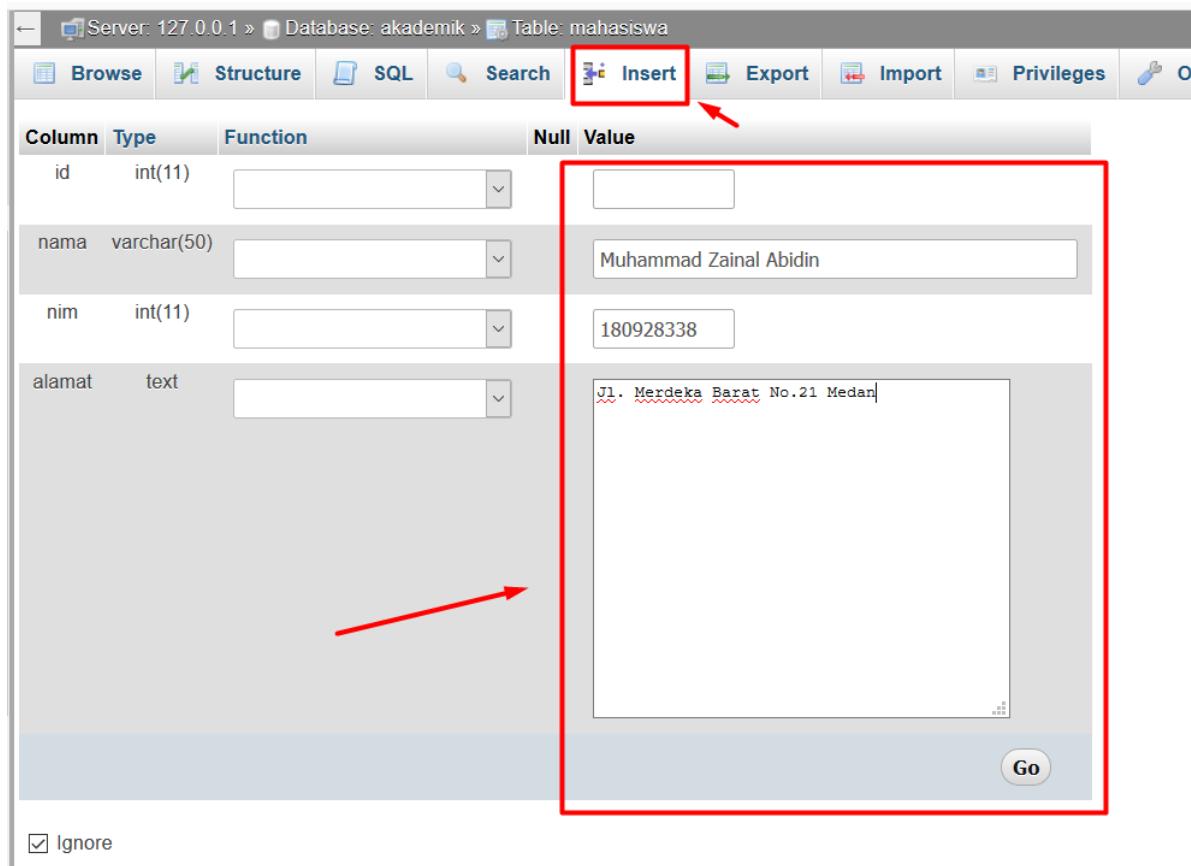
A screenshot of the MySQL Workbench interface, showing the 'Structure' tab for a table named 'mahasiswa'. The 'Table name' field contains 'mahasiswa', which is highlighted with a red box. The table structure is defined with four columns: 'id', 'nama', 'nim', and 'alamat'. The 'id' column is set to type 'INT', 'nama' to 'VARCHAR(50)', 'nim' to 'INT', and 'alamat' to 'TEXT'. In the 'Attributes' section, the 'id' column has its 'Null' checkbox unchecked and its 'Index' checkbox checked, with 'PRIMARY' selected. The 'A.I.' (Auto Increment) checkbox is also checked for the 'id' column. A red box highlights the 'id' column and its primary key settings.

membuat table mahasiswa

id kita buat bertipe int, nama varchar dengan nilai 50, nim bertipe int dan alamat bertipe text. jangan lupa menentukan *primary key* nya pada id, dan centang *auto increment* nya.

Sampai di sini kita telah selesai membuat tabel *mahasiswa* dalam database *akademik*.

sekarang coba input data mahasiswa, dengan klik pada menu insert.



menginput data ke database mysql

Oke kita selesai dengan persiapannya.

## Koneksi Database PHP dan MySQLi

Untuk membuat CRUD, tahap awal yang kita lakukan adalah membuat file koneksi nya dulu, baru kemudian kita menampilkan data dari database. jadi pada tutorial ini kita akan belajar sampai menampilkan data dari database dengan PHP dan MySQLi. pada tutorial selanjutnya baru kita akan melanjutkan dengan membahas cara membuat input data ke database.

Pertama kita buat dulu folder baru pada folder htdocs kita. di sini saya membuat sebuah folder baru dengan nama "*akademik*". karena kita akan mencoba membuat CRUD data mahasiswa sebagai contoh.

Setelah membuat folder akademik. kemudian buat sebuah file php baru dengan nama "*koneksi.php*".

oh ya di sini saya tidak lagi menjelaskan lagi dengan detail tentang cara membuat koneksi database PHP dengan MySQLi. karena sudah kita jelaskan pada tutorial sebelumnya. silahkan baca tutorial membuat koneksi database PHP dan MySQLi pada tutorial sebelumnya.

ketikkan syntax koneksi berikut pada file koneksi.php

### koneksi.php

```
<?php
$koneksi = mysqli_connect("localhost","root","","akademik");

// Check connection
if (mysqli_connect_errno()){
    echo "Koneksi database gagal : " . mysqli_connect_error();
}

?>
```

setelah membuat koneksi, selanjutnya kita membuat file php yang bertugas untuk menampilkan data mahasiswa dari database dengan mysqli.

### Menampilkan Data Dari Database

Buat sebuah file php baru dengan nama *index.php* dalam folder *akademik*. pada *index.php* ini kita akan menampilkan data mahasiswa.

### index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>CRUD PHP dan MySQLi - WWW.MALASNGODING.COM</title>
</head>
<body>

    <h2>CRUD DATA MAHASISWA - WWW.MALASNGODING.COM</h2>
    <br/>
    <a href="tambah.php">+ TAMBAH MAHASISWA</a>
    <br/>
    <br/>
    <table border="1">
        <tr>
            <th>NO</th>
            <th>Nama</th>
            <th>NIM</th>
```

```

<th>Alamat</th>
<th>OPSI</th>
</tr>
<?php
include 'koneksi.php';
$no = 1;
$data = mysqli_query($koneksi,"select * from mahasiswa");
while($d = mysqli_fetch_array($data)){
    ?>
    <tr>
        <td><?php echo $no++; ?></td>
        <td><?php echo $d['nama']; ?></td>
        <td><?php echo $d['nim']; ?></td>
        <td><?php echo $d['alamat']; ?></td>
        <td>
            <a href="edit.php?id=<?php echo $d['id']; ?>">EDIT</a>
            <a href="hapus.php?id=<?php echo $d['id']; ?>">HAPUS</a>
        </td>
    </tr>
    <?php
}
?>
</table>
</body>
</html>

```

seperti pada gambar berikut

```
index.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>CRUD PHP dan MySQLi - WWW.MALASNGODING.COM</title>
5 </head>
6 <body>
7
8     <h2>CRUD DATA MAHASISWA - WWW.MALASNGODING.COM</h2>
9     <br/>
10    <a href="tambah.php">+ TAMBAH MAHASISWA</a>
11    <br/>
12    <br/>
13    <table border="1">
14        <tr>
15            <th>NO</th>
16            <th>Nama</th>
17            <th>NIM</th>
18            <th>Alamat</th>
19            <th>OPSI</th>
20        </tr>
21        <?php
22            include 'koneksi.php';
23            $no = 1;
24            $data = mysqli_query($koneksi,"select * from mahasiswa");
25            while($d = mysqli_fetch_array($data)){
26                ?>
27                <tr>
28                    <td><?php echo $no++; ?></td>
29                    <td><?php echo $d['nama']; ?></td>
30                    <td><?php echo $d['nim']; ?></td>
31                    <td><?php echo $d['alamat']; ?></td>
32                    <td>
33                        <a href="edit.php?id=<?php echo $d['id']; ?>">EDIT</a>
34                        <a href="hapus.php?id=<?php echo $d['id']; ?>">HAPUS</a>
35                    </td>
36                </tr>
37            <?php
38        }
39    ?>
40    </table>
41 </body>
42 </html>
```

membuat crud dengan php dan mysqli – menampilkan data

kemudian, jalankan pada browser dengan alamat "*localhost/akademik/*"  
maka hasilnya sebagai berikut.



## CRUD DATA MAHASISWA - WWW.MALASNGODING.COM

[+ TAMBAH MAHASISWA](#)

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>

menampilkan data dari database dengan php dan mysqli

nah, data yang sebelumnya telah kita input di table mahasiswa pun berhasil kita tampilkan.

Sebenarnya sama saja dengan mysql. hanya bedanya di sini kita menambahkan i pada setiap fungsi mysql. sehingga menjadi "mysqli". seperti yang teman-teman lihat pada syntax di index.php yang telah kita buat di atas. mysql\_query() menjadi mysqli\_query(). kemudian di ikuti dengan menambahkan variabel **\$koneksi** pada parameter pertama. *mysqli\_query(\$koneksi,"select \* from mahasiswa");* . dari mana asal variabel \$koneksi ini?

yaitu dari file koneksi.php yang telah kita buat. perhatikan pada file koneksi.php

```
$koneksi = mysqli_connect("localhost","root","","akademik");
```

nah, cukup mudah kan teman-teman. selanjutnya akan kita lanjutkan pada tutorial membuat CRUD dengan PHP dan MySQLi nya pada part selanjutnya. yaitu kita kan melanjutkan link tambah mahasiswa, edit dan hapus seperti contoh di atas.

## Menginput Data Ke Database

Pada tutorial ini kita akan melanjutkan tutorial sebelumnya, yaitu kita akan membuat proses input data ke database dengan PHP dan MySQLi. Pada tutorial sebelumnya kita telah belajar cara membuat koneksi PHP dengan Database menggunakan MySQLi dan cara menampilkan data dari database dengan PHP menggunakan MySQLi.

Yuk langsung saja di simak tentang kelanjutannya pada tutorial ini. di sini kita akan belajar cara menginput data ke database dengan PHP menggunakan MySQLi.

Saya sarankan untuk teman-teman yang baru belajar, silahkan baca dulu tutorial sebelumnya tentang pengelolaan form (form handling). karena pada tutorial input data ke database ini kita akan bermain-main dengan method POST.

Tahap pertama, silahkan teman-teman perhatikan pada file index.php yang telah kita buat sebelumnya.

pada file index.php, kita telah membuat sebuah link “tambah mahasiswa”. yang mengarahkan ke halaman tambah.php jika di klik.

```
<a href="tambah.php">+ TAMBAH MAHASISWA</a>
```

oleh sebab itu kita harus membuat file *tambah.php*. pada file tambah.php ini kita akan membuat form penginputan data mahasiswa. yang mana pada saat tombol simpannya di klik, data mahasiswa yang di input akan tersimpan ke database.

buat sebuah file file baru dengan nama *tambah.php*.

**tambah.php**

```
<!DOCTYPE html>
<html>
<head>
    <title>CRUD PHP dan MySQLi - WWW.MALASNGODING.COM</title>
</head>
<body>

    <h2>CRUD DATA MAHASISWA - WWW.MALASNGODING.COM</h2>
    <br/>
    <a href="index.php">KEMBALI</a>
    <br/>
    <br/>
```

```

<h3>TAMBAH DATA MAHASISWA</h3>
<form method="post" action="tambah_aksi.php">
    <table>
        <tr>
            <td>Nama</td>
            <td><input type="text" name="nama"></td>
        </tr>
        <tr>
            <td>NIM</td>
            <td><input type="number" name="nim"></td>
        </tr>
        <tr>
            <td>Alamat</td>
            <td><input type="text" name="alamat"></td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" value="SIMPAN"></td>
        </tr>
    </table>
</form>
</body>
</html>

```

Sekarang jalankan pada browser kembali. dan klik pada “+ tambah mahasiswa”. maka halaman akan di alihkan ke form tambah mahasiswa yang telah kita buat pada file tambah.php.



## CRUD DATA MAHASISWA - WWW.MALASNGODING.COM

[KEMBALI](#)

### TAMBAH DATA MAHASISWA

Nama

NIM

Alamat



membuat crud dengan php dan mysqli – input data

sampai di sini kita belum selesai, kita cuma baru selesai membuat form penginputan data mahasiswa nya saja. tapi belum membuat aksi pada saat tombol “simpan” di klik.

Perhatikan lagi pada action form yang telah kita buat pada file tambah.php di atas.

```
<form method="post" action="tambah_aksi.php">
```

Kita menetapkan aksi nya di file *tambah\_aksi.php*. dengan method pengiriman data menggunakan POST.

karena action form ini akan di jalankan pada file *tambah\_aksi.php*, maka kita harus membuat file *tambah\_aksi.php* terlebih dulu. buat file baru dengan nama *tambah\_aksi.php*.

*tambah\_aksi.php*

```
<?php  
// koneksi database  
include 'koneksi.php';  
  
// menangkap data yang di kirim dari form  
$nama = $ POST['nama'];
```

```
$nim = $_POST['nim'];
$alamat = $_POST['alamat'];

// menginput data ke database
mysqli_query($koneksi,"insert into mahasiswa
values('','$nama','$nim','$alamat')");

// mengalihkan halaman kembali ke index.php
header("location:index.php");

?>
```

Coba teman-teeman perhatikan lagi pada file *tambah\_aksi.php* di atas. pertama kita menangkap data yang dikirim dari form. kemudian kita input datanya ke database dengan fungsi *mysqli\_query()* :

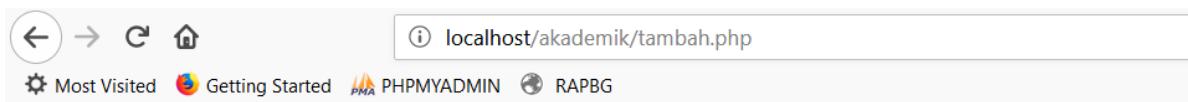
```
mysqli_query($koneksi,"insert into mahasiswa
values('','$nama','$nim','$alamat')");
```

dan terakhir setelah data di input ke database, selanjutnya kita alihkan kembali halamannya ke file *index.php*.

```
// mengalihkan halaman kembali ke index.php
header("location:index.php");
```

oh ya jangan lupa juga, karena pada aksi ini kita akan berhubungan dengan database, maka pada awal baris nya jangan lupa di include file *koneksi.php* nya, seperti yang telah di jelaskan pada tutorial sebelumnya.

Sekarang mari kita coba jalankan kembali. dan input data mahasiswa barunya.



[KEMBALI](#)

### TAMBAH DATA MAHASISWA

Nama	<input type="text" value="Junaidi"/>
NIM	<input type="text" value="1202833933"/>
Alamat	<input type="text" value="jl pinang raya no 171"/>

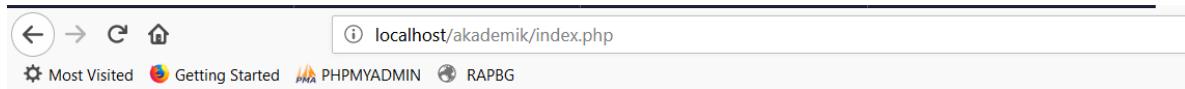
SIMPAN



menginput data ke database dengan php dan mysqli

dan kemudian klik tombol simpan. untuk mengimpan data ke database.

maka hasilnya seperti berikut.



#### CRUD PHP MYSQLI

nah, data yang diinput pun telah berhasil di simpan ke database dan tampil secara otomatis.

sampai di sini selesailah sudah proses penginputan data ke database dengan php dan menggunakan MySQLi.

#### Edit & Update Data Pada Database

Pada tutorial membuat CRUD dengan PHP dan MySQLi sebelumnya kita telah membahas tentang cara menampilkan data dari database dan cara menginput data ke database. semoga teman-teman telah bisa menguasai materi sebelumnya yang telah disebutkan tadi. Karena pada tutorial ini, kita akan belajar cara mengedit atau mengupdate data dari database dengan PHP dan MySQLi.

Seperti yang bisa teman-teman lihat pada file index.php yang menampilkan data mahasiswa, pada masing-masing row kita telah membuat link edit dan link hapus, yang mengirimkan data menggunakan method GET.

Saya sarankan untuk teman-teman yang belum tau apa itu method atau cara penanganan form dengan php, silahkan teman-teman baca lagi tentang tutorial form handling di PHP.

Perhatikan pada link edit yang telah kita buat pada file index.php

```
<a href="edit.php?id=<?php echo $d['id']; ?>">EDIT</a>
```

link akan kita mengarah ke file edit.php saat di klik. sambil mengirim data id.

kemudian buat sebuah file baru dengan nama edit.php. pada file edit.php ini kita akan membuat form edit untuk data mahasiswa.

edit.php

```
<!DOCTYPE html>
<html>
<head>
    <title>CRUD PHP dan MySQLi - WWW.MALASNGODING.COM</title>
</head>
<body>

    <h2>CRUD DATA MAHASISWA - WWW.MALASNGODING.COM</h2>
    <br/>
    <a href="index.php">KEMBALI</a>
    <br/>
    <br/>
    <h3>EDIT DATA MAHASISWA</h3>

    <?php
    include 'koneksi.php';
    $id = $_GET['id'];
    $data = mysqli_query($koneksi,"select * from mahasiswa where id='$id'");
    while($d = mysqli_fetch_array($data)){
        ?>
        <form method="post" action="update.php">
            <table>
                <tr>
                    <td>Nama</td>
                    <td>
                        <input type="hidden" name="id" value="<?php echo $d['id']; ?>">
                        <input type="text" name="nama" value="<?php echo $d['nama']; ?>">
                    </td>
                </tr>
                <tr>
                    <td>NIM</td>
                    <td><input type="number" name="nim" value="<?php echo $d['nim']; ?>"></td>
                </tr>
            </table>
        </form>
    }
</body>
```

```

        </tr>
        <tr>
            <td>Alamat</td>
            <td><input type="text" name="alamat" value=<?php echo $d['alamat']; ?>"></td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" value="SIMPAN"></td>
        </tr>
    </table>
</form>
<?php
}
?>

</body>
</html>

```

Jika teman-teman telah membaca tutorial tentang form handling sebelumnya pasti teman-teman telah mengerti. pada file edit.php ini kita tangkap id yang di kirim melalui url (GET). kemudian kita ambil data mahasiswa yang memiliki id tersebut untuk kita edit.

```

$id = $_GET['id'];
$data = mysqli_query($koneksi,"select * from mahasiswa where id='$id'");
while($d = mysqli_fetch_array($data)){

```

kemudian kita tampilkan di dalam form.

```

<form method="post" action="update.php">
    <table>
        <tr>
            <td>Nama</td>
            <td>
                <input type="hidden" name="id" value=<?php echo
$id['id']; ?>">
                <input type="text" name="nama" value=<?php echo
$d['nama']; ?>">
            </td>
        </tr>
        <tr>
            <td>NIM</td>
            <td><input type="number" name="nim" value=<?php echo
$d['nim']; ?>"></td>
        </tr>
        <tr>
            <td>Alamat</td>
            <td><input type="text" name="alamat" value=<?php echo
$d['alamat']; ?>"></td>
        </tr>
    </table>
</form>

```

```

<tr>
    <td></td>
    <td><input type="submit" value="SIMPAN"></td>
</tr>
</table>
</form>

```

pada form edit ini kita juga membuat data id pada form yang kita sembunyikan.

```
<input type="hidden" name="id" value="php echo $d['id']; ?&gt;"&gt;</pre

```

ini berguna untuk menyimpan data id mahasiswa yang sedang di edit. sehingga pada saat data ini di update, data mahasiswa yang di update adalah data mahasiswa yang memiliki id tersebut.

saya yakin teman-teeman pasti mudah memahaminya, jika masih bingung langsung saja teman-teaman praktekkan dan bereksperimen.

coba kita lihat hasilnya dulu seperti apa. jalankan pada browser dan klik edit.

## **CRUD DATA MAHASISWA - WWW.MALASNGODING.COM**

[+ TAMBAH MAHASISWA](#)

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>
2	Junaidi	2147483647	jl pinang raya no 171	<a href="#">EDIT</a> <a href="#">HAPUS</a>



update data

di sini saya akan mencoba mengedit data mahasiswa yang bernama junaidi. saat di klik pada edit. maka hasilnya akan seperti berikut.



### KEMBALI

#### **EDIT DATA MAHASISWA**

Nama	<input type="text" value="Junaidi"/>
NIM	<input type="text" value="2147483647"/> <input type="button" value="▼"/>
Alamat	<input type="text" value="jl pinang raya no 171"/>
<input type="button" value="SIMPAN"/>	

edit data dengan php dan mysqli

nah, selesai sampai tahap edit. sekarang kita akan membuat aksi dari form edit ini. seperti yang teman-teman lihat pada tag form "<form>". action nya telah kita tetapkan ke file update.php

karena kita ingin data mahasiswa yang ingin di edit ini di proses pada file update.php.

oleh sebab itu buat satu buah file lagi dengan nama update.php.

**update.php**

```
<?php
// koneksi database
include 'koneksi.php';

// menangkap data yang di kirim dari form
$id = $_POST['id'];
$nama = $_POST['nama'];
$nim = $_POST['nim'];
$alamat = $_POST['alamat'];

// update data ke database
mysqli_query($koneksi,"update mahasiswa set nama='$nama', nim='$nim',
alamat='$alamat' where id='$id'");

// mengalihkan halaman kembali ke index.php
header("location:index.php");
```

```
?>
```

di sini pertama kita hubungkan dulu dengan file koneksi.

```
include 'koneksi.php';
```

kemudian kita tangkap data yang di kirim dari form edit.

```
$id = $_POST['id'];
$nama = $_POST['nama'];
$nim = $_POST['nim'];
$alamat = $_POST['alamat'];
```

kemudian kita update ke database dengan query berikut.

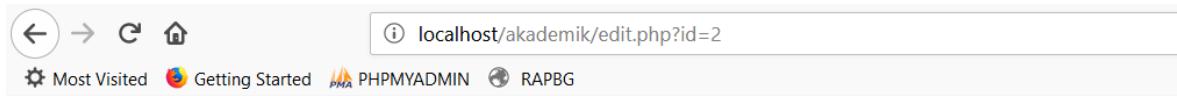
```
mysqli_query($koneksi,"update mahasiswa set nama='$nama', nim='$nim',
alamat='$alamat' where id='$id'");
```

dan terakhir setelah data berhasil di update. halaman akan di alihkan kembali ke file index.php atau halaman depan kita.

```
header("location:index.php");
```

Sekarang mari kita coba, apakah berhasil?

Jalankan pada browser dan klik edit. di sini saya akan mencoba mengedit data mahasiswa yang bernama “junaidi”. dan saya mengubah namanya menjadi “junaidi abadi”. dan klik tombol simpan.



## CRUD DATA MAHASISWA - WWW.MALASNGODING.COM

[KEMBALI](#)

### EDIT DATA MAHASISWA

Nama

NIM

Alamat

membuat edit dan update database dengan php dan mysqli

klik tombol simpan. maka data akan di update.

## CRUD DATA MAHASISWA - WWW.MALASNGODING.COM

[+ TAMBAH MAHASISWA](#)

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>
2	Junaidi Abadi	2147483647	jl pinang raya no 171	<a href="#">EDIT</a> <a href="#">HAPUS</a>

berhasil mengupdate data

nah data junaidi pun berhasil di ubah. dari *junaidi* menjadi *junaidi abadi*.

Oke selesai sudah proses edit data dengan PHP dan MySQLi. pada tutorial selanjutnya kita akan membuat hapus data dari database dengan PHP dan MySQLi.

## Hapus Data Dari Database

Silahkan teman-teman perhatikan pada file index.php. pada file index.php kita membuat link atau tombol untuk menghapus data yang di pilih. dengan sekalian mengirimkan id ke berapa yang ingin di hapus pada parameter id nya.

```
<a href="hapus.php?id=<?php echo $d['id']; ?>">HAPUS</a>
```

pada saat tombol atau link hapus tersebut di klik, maka halaman akan di alihkan ke file hapus.php, sambil mengirim data id menggunakan GET.

sekarang kita buat sebuah file baru dengan nama hapus.php.

hapus.php

```
<?php
// koneksi database
include 'koneksi.php';

// menangkap data id yang di kirim dari url
$id = $_GET['id'];

// menghapus data dari database
mysqli_query($koneksi,"delete from mahasiswa where id='$id'");

// mengalihkan halaman kembali ke index.php
header("location:index.php");

?>
```

coba teman-teman perhatikan di sini. pada file hapus.php.

langkah pertama jangan pernah lupa untuk meng include file koneksi database jika kita akan menggunakan pengolahan database. selanjutnya kita menangkap data id nya yang dikirim dengan GET. lalu kita masukkan dalam variabel \$id.

selanjutnya langsung saja kita hapus data mahasiswa yang ber id tersebut (sesuai id yang di hapus). terakhir halaman akan di alihkan kembali ke index.php setelah data mahasiswa di hapus.

sangat mudah bukan. saya yakin teman-teman bisa memahaminya.

sekarang coba kita jalankan kembali project CRUD mahasiswa yang kita buat. "localhost/akademik".

The screenshot shows a web browser window with the URL `localhost/akademik/`. The page title is **CRUD DATA MAHASISWA - WWW.MALASNGODING.COM**. Below the title is a link [+ TAMBAH MAHASISWA](#). A table displays student information:

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>
2	Junaidi Abadi	2147483647	jl pinang raya no 171	<a href="#">EDIT</a> <a href="#">HAPUS</a>

A red arrow points to the [HAPUS](#) button for the second row (Junaidi Abadi).

membuat crud dengan php dan mysqli hapus data

klik hapus. di sini saya mencoba klik hapus pada data mahasiswa yang bernama junaidi. maka hasilnya.

The screenshot shows a web browser window with the URL `localhost/akademik/index.php`. The page title is **CRUD DATA MAHASISWA - WWW.MALASNGODING.COM**. Below the title is a link [+ TAMBAH MAHASISWA](#). A table displays student information:

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>

**CRUD DATA MAHASISWA - WWW.MALASNGODING.COM**

[+ TAMBAH MAHASISWA](#)

NO	Nama	NIM	Alamat	OPSI
1	Muhammad Zainal Abidin	180928338	Jl. Merdeka Barat No.21 Medan	<a href="#">EDIT</a> <a href="#">HAPUS</a>

menghapus data dari database dengan php dan mysqli

Nah sampai di sini kita telah berhasil menghapus data dari database dengan PHP dan MySQLi. sangat mudah bukan. silahkan teman-teman coba sendiri.

## Membuat Login Dengan PHP dan MySQLi

Sebelumnya di ebook ini kita telah membahas tentang cara membuat login dengan menggunakan PHP dan MySQL (tanpa i) . maka kali ini kita akan membahas tentang tutorial cara membuat login dengan PHP dan MySQLi.

Untuk teman-teman yang ingin mempelajari juga cara membuat login dengan menggunakan PHP dan MySQL, teman-teman bisa membacanya pada sub judul sebelumnya.

Jika teman-teman sudah pernah membaca tutorial sebelumnya tentang cara membuat login di [www.malasngoding.com](http://www.malasngoding.com) (link yang tertera di atas), pasti teman-teman tidak akan kesulitan lagi untuk mengikuti tutorial membuat login dengan PHP dan MySQLi ini. karena intinya sama saja, yang berbeda hanya penggunaan fungsi mysql dan mysqli. yuk langsung simak saja.

### Persiapan

1 . Untuk memulai tutorial dan project membuat login dengan PHP dan MySQLi seperti di contoh ini, silahkan buat sebuah folder baru dengan nama login pada direktori localhost teman-teman.

2 . Buat sebuah database baru dengan nama akademik. kemudian buat sebuah tabel baru dengan nama admin dalam database akademik. isi dari tabel admin yaitu id, username dan password. dimana id bertipe int, pilih primary dan centang

auto\_increment nya. username dan password sama-sama bertipe varchar dengan nilai 225.

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.	Comments	Virtual
id	INT			None				PRIMARY	PRIMARY	
username	VARCHAR	255		None						
password	VARCHAR	255		None						

Membuat table admin

dan selesai.

selanjutnya isikan data admin. untuk data contoh saja. untuk contoh akun admin yang akan login. isikan username nya 'admin', dan password nya 'admin123'.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	<span>Change</span> <span>Drop</span> <span>Primary</span> <span>Unique</span> <span>Index</span> <span>Spatial</span> <span>Fulltext</span> <span>Distinct values</span> <span>More</span>
2	username	varchar(255)			No	None		<span>Change</span> <span>Drop</span> <span>Primary</span> <span>Unique</span> <span>Index</span> <span>Spatial</span> <span>Fulltext</span> <span>Distinct values</span> <span>More</span>
3	password	varchar(255)			No	None		<span>Change</span> <span>Drop</span> <span>Primary</span> <span>Unique</span> <span>Index</span> <span>Spatial</span> <span>Fulltext</span> <span>Distinct values</span> <span>More</span>

Tabel admin

Server: 127.0.0.1 » Database: akademik » Table: admin

Column	Type	Function	Null	Value
id	int(11)			
username	varchar(255)			admin
password	varchar(255)			admin123

→

Insert data admin

Server: 127.0.0.1 » Database: akademik » Table: admin

Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)

```
SELECT * FROM `admin`
```

Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

	id	username	password
<input type="checkbox"/>	1	admin	admin123

Check all | With selected:  Edit  Copy  Delete  Export

Data username dan password admin

persiapan di database selesai.

## Membuat Koneksi Database

Sudah sering disinggung, jika kita ingin bermain-main dengan database, kita wajib untuk membuat koneksi database dulu. tujuannya agar Project PHP yang kita buat bisa terhubung dengan database.

Untuk cara membuat koneksi database dengan PHP dan MySQLi, saya telah membahas nya pada tutorial sebelumnya. silahkan teman-teman baca dulu tutorial cara membuat koneksi database dengan php dan mysqli nya dengan klik pada link di atas, karena penjelasannya sudah lengkap dan di sini tidak lagi di jelaskan tentang membuat koneksi database dengan detail.

Buat sebuah file baru dengan nama koneksi.php dan simpan pada folder login yang telah kita buat.

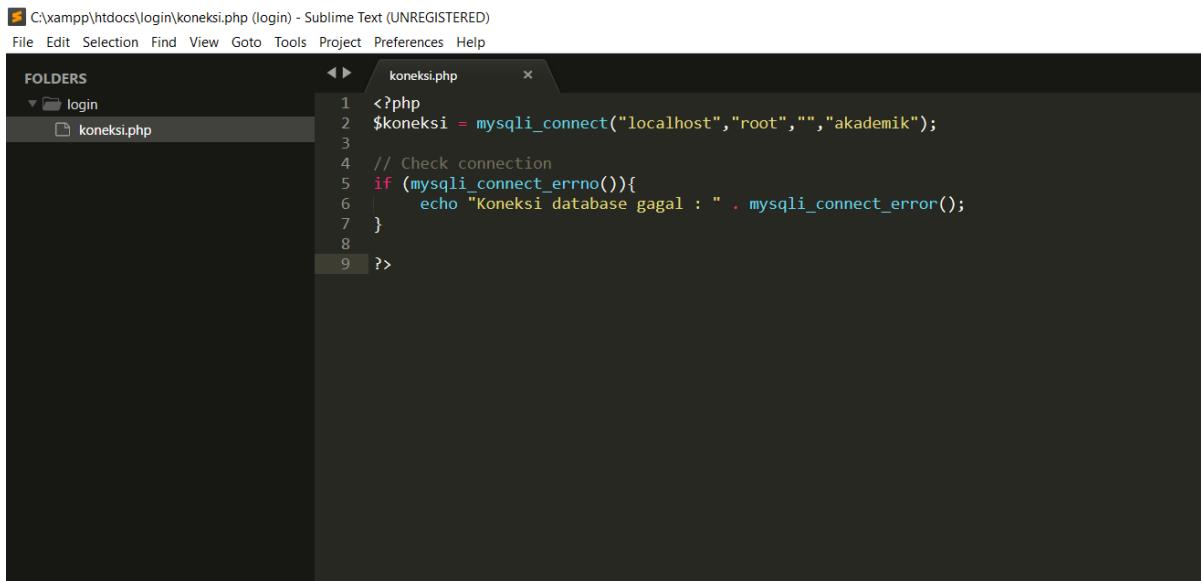
koneksi.php

```
<?php
$koneksi = mysqli_connect("localhost","root","","akademik");

// Check connection
if (mysqli_connect_errno()){
    echo "Koneksi database gagal : " . mysqli_connect_error();
}

?>
```

Silahkan teman-teman isi sendiri nama database nya sesuai nama database yang teman-teman buat, karena di sini kita akan menggunakan database akademik, maka nama databasenya akademik.



```
C:\xampp\htdocs\login\koneksi.php (login) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
    login
        koneksi.php
koneksi.php
1 <?php
2 $koneksi = mysqli_connect("localhost","root","","akademik");
3
4 // Check connection
5 if (mysqli_connect_errno()){
6     echo "Koneksi database gagal : " . mysqli_connect_error();
7 }
8
9 ?>
```

membuat koneksi database dengan php dan mysqli

sampai di sini koneksi database php dan mysqli telah selesai kita buat.

Selanjutnya akan kita buat halaman login sederhana untuk tempat admin memasukkan username dan password. kemudian di alihkan ke halaman admin.

## Membuat Login PHP dan MySQLi

Sekarang kita akan melanjatkannya dengan membuat halaman login sederhana untuk admin. selanjutnya akan di jelaskan secara bertahap agar lebih mudah di pahami oleh teman-teman.

buat sebuah file baru lagi dengan nama *index.php*.

*index.php*

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Login Dengan PHP dan MySQLi -
    WWW.MALASNGODING.COM</title>
</head>
<body>
    <h2>Login - www.malasngoding.com</h2>
    <br/>
    <!-- cek pesan notifikasi -->
    <?php
        if(isset($_GET['pesan'])) {
            if($_GET['pesan'] == "gagal") {
                echo "Login gagal! username dan password salah!";
            }
        }
    ?>
```

```

} else if($_GET['pesan'] == "logout"){
    echo "Anda telah berhasil logout";
} else if($_GET['pesan'] == "belum_login"){
    echo "Anda harus login untuk mengakses halaman admin";
}
}
?>
<br/>
<br/>
<form method="post" action="cek_login.php">
    <table>
        <tr>
            <td>Username</td>
            <td>:</td>
            <td><input type="text" name="username"
placeholder="Masukkan username"></td>
        </tr>
        <tr>
            <td>Password</td>
            <td>:</td>
            <td><input type="password" name="password"
placeholder="Masukkan password"></td>
        </tr>
        <tr>
            <td></td>
            <td></td>
            <td><input type="submit" value="LOGIN"></td>
        </tr>
    </table>
</form>
</body>
</html>

```

Kita lanjutkan lagi dengan membuat file yang akan mengelola data yang di kirim dari form login dari index.php.

Pada file index.php yang membuat halaman login di atas kita telah menetapkan aksi dari form ini adalah '*cek\_login.php*' dengan method 'post' untuk jenis pengiriman datanya. silahkan baca lagi tentang **GET** dan **POST** jika teman-teman belum mempelajarinya. sudah di bahas juga di pembahasan sebelumnya.

buat sebuah file baru lagi dengan nama *cek\_login.php*.

### cek\_login.php

```

<?php
// mengaktifkan session php
session_start();

```

```
// menghubungkan dengan koneksi
include 'koneksi.php';

// menangkap data yang dikirim dari form
$username = $_POST['username'];
$password = $_POST['password'];

// menyeleksi data admin dengan username dan password yang sesuai
$data = mysqli_query($koneksi,"select * from admin where username='$username'
and password='$password'");

// menghitung jumlah data yang ditemukan
$cek = mysqli_num_rows($data);

if($cek > 0){
    $_SESSION['username'] = $username;
    $_SESSION['status'] = "login";
    header("location:admin/index.php");
} else{
    header("location:index.php?pesan=gagal");
}
?>
```

Perhatikan, di sini kita aktifkan dulu session php. karena kita menggunakan session sebagai penanda status pengguna yang akan login.

kita mengaktifkan session php dengan syntax berikut

```
session_start();
```

Kemudian jangan lupa juga untuk menghubungkannya dengan file koneksi database yang sudah kita buat.

```
include 'koneksi.php';
```

kemudian menangkap data yang dikirim dari form login.

```
$username = $_POST['username'];
$password = $_POST['password'];
```

Kemudian kita seleksi data admin yang memiliki username dan password yang sesuai dengan yang di input pada form login.

```
$data = mysqli_query($koneksi,"select * from admin where username='$username'
and password='$password'");
```

Selanjutnya kita hitung jumlah record yang sesuai (yang sesuai dengan username dan password yang di input).

```
$cek = mysqli_num_rows($data);
```

Jika data username dan password admin di temukan di tabel ‘admin’, maka nilai dari variabel \$cek adalah jumlah dari record yang di temukan. berarti variabel \$cek bernilai 1 jika username dan password yang di input sesuai dengan yang ada di database. karena fungsi mysqli\_num\_rows() berfungsi untuk menghitung jumlah record hasil dari query yang ada dalam parameter si fungsi mysqli\_num\_rows() ini.

Selanjutnya kita cek dengan if else jika variabel \$cek lebih besar dari nol (berarti username dan password yang di input sesuai), maka kita buat session username dan session status seperti pada syntax berikut. kemudian kita alihkan ke folder admin/index.php.

Jika tidak maka kita alihkan halamannya kembali ke halaman login (index.php) sambil mengirimkan data pesan (?pesan=gagal).

```
if($cek > 0){  
    $_SESSION['username'] = $username;  
    $_SESSION['status'] = "login";  
    header("location:admin/index.php");  
}else{  
    header("location:index.php?pesan=gagal");  
}
```

itu sebabnya di file index.php (halaman login) kita membuat pengecekan seperti berikut (perhatikan pada file index.php)

```
<?php  
if(isset($_GET['pesan'])){  
    if($_GET['pesan'] == "gagal"){  
        echo "Login gagal! username dan password salah!";  
    }else if($_GET['pesan'] == "logout"){  
        echo "Anda telah berhasil logout";  
    }else if($_GET['pesan'] == "belum_login"){  
        echo "Anda harus login untuk mengakses halaman admin";  
    }  
?>
```

Ini untuk memeriksa pesan yang di kirim. karena tadi jika username dan password tidak di temukan, akan dikirimkan pesan "gagal". pada pengecekan notifikasi ini kita akan menampilkan pesan notifikasi "Login gagal! username dan password salah!" jika username dan password gagal. begitu juga seterusnya dengan pesan logout dan belum\_login yang akan kita buat pengalihannya sebentar lagi.

## Membuat Halaman Admin

Seperti yang teman-teman lihat pada syntax sebelumnya. saat melakukan pengecekan, jika username dan password benar, maka halaman akan kita alihkan ke halaman admin yang berada dalam folder admin.

jadi sekarang kita buat sebuah folder baru dengan nama 'admin'.

ODE > Local Disk (C:) > xampp > htdocs > login >				
Name	Date modified	Type	Size	
admin	3/19/2018 11:02	File folder		
cek_login	3/19/2018 17:25	PHP File	1 KB	
index	3/19/2018 11:25	PHP File	2 KB	
koneksi	3/18/2018 19:19	PHP File	1 KB	

membuat login dengan php dan mysqli

Setelah membuat folder admin, buat sebuah file baru lagi dengan nama index.php. pada file index.php ini kita akan membuat halaman sederhana untuk admin.

Buat sebuah file baru dengan nama index.php dalam folder admin.

admin/index.php

```
<!DOCTYPE html>
<html>
<head>
<title>Membuat Login Dengan PHP dan MySQLi -
```

```

WWW.MALASNGODING.COM</title>
</head>
<body>
    <h2>Halaman Admin</h2>

    <br/>

    <!-- cek apakah sudah login -->
    <?php
    session_start();
    if($_SESSION['status']!="login") {
        header("location:../index.php?pesan=belum_login");
    }
    ?>

    <h4>Selamat datang, <?php echo $_SESSION['username']; ?>! anda telah
login.</h4>

    <br/>
    <br/>

    <a href="logout.php">LOGOUT</a>

</body>
</html>

```

coba perhatikan pada file index.php. di sini kita buat pengecekan dengan if else.

```

<!-- cek apakah sudah login -->
<?php
session_start();
if($_SESSION['status']!="login") {
    header("location:../index.php?pesan=belum_login");
}
?>

```

Jika session status tidak sama dengan “login” maka halaman akan di alihkan ke halaman login lagi sambil mengirim pesan “belum\_login”. pengecekan ini berfungsi untuk keamanan agar halaman admin tidak bisa di akses sebelum melakukan login (session login di buat saat login berhasil, perhatikan lagi pengecekan pada file cek\_login.php).

Kita juga menampilkan notifikasi selamat datang untuk username yang sedang login dengan cara langsung menampilkan session username yang berhasil login. jika yang login adalah pengguna dengan username “admin” maka yang tampil di sini adalah “selamat datang, admin”. begitu juga selainnya.

terakhir kita buat tombol untuk link logout.

```
<a href="logout.php">LOGOUT</a>
```

saat link logout di klik, maka akan di alihkan ke file logout.php.

buat file baru dengan nama logout.php dalam folder admin.

admin/logout.php

```
<?php
// mengaktifkan session
session_start();

// menghapus semua session
session_destroy();

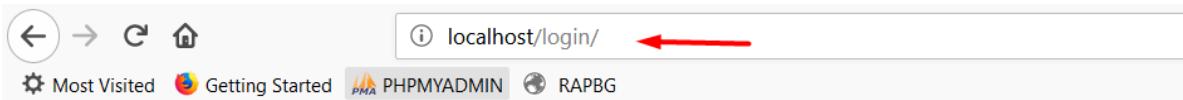
// mengalihkan halaman sambil mengirim pesan logout
header("location:../index.php?pesan=logout");
?>
```

file logout berguna untuk menghapus semua session yang telah dibuat saat login. dan mengalihkan halaman kembali ke halaman login sambil mengirim pesan 'logout'.

kita menghapus semua session dengan fungsi session\_destroy() seperti pada contoh di atas.

nah, semuanya selesai. waktunya kita coba.

jalankan localhost/login pada browser.



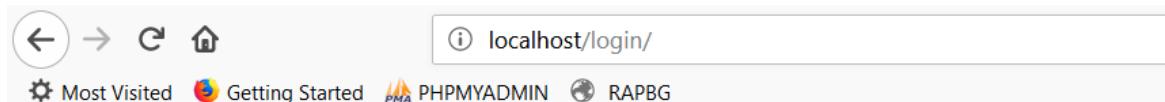
## Login - www.malasngoding.com

Username :  ←

Password :  ←

membuat login dengan php dan mysqli part 2

nah, halaman login tampil. selanjutnya untuk mencoba apakah form login ini sudah berjalan dengan sebagai mana mestinya, coba isikan username : tes, dan password tes123.



## Login - www.malasngoding.com

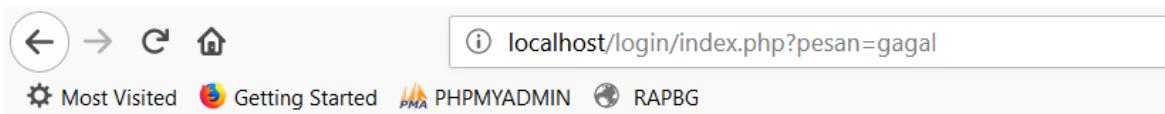
Username :  ←

Password :  ←

←

contoh login dengan php dan mysqli

dan klik tombol LOGIN. maka akan muncul notifikasi kalau username dan password yang di masukkan salah.



## Login - www.malasngoding.com

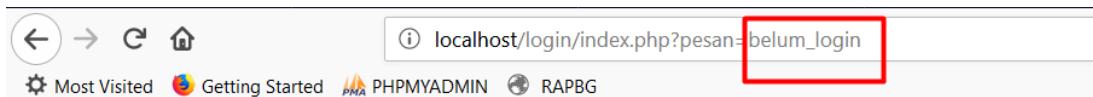
Login gagal! username dan password salah!

Username :

Password :

login gagal

sekarang coba masuk langsung ke halaman admin tanpa melakukan login, akses "localhost/login/admin".



Anda harus login untuk mengakses halaman admin

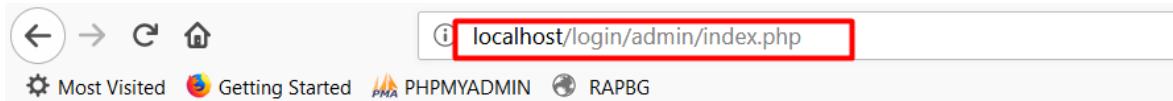
Username :

Password :

keamanan login admin php dan mysqli

maka akan muncul notifikasi yang mengharuskan kita untuk login terlebih dulu.

mantap. sekarang coba masukkan username dan password yang benar. username : admin, password : admin123. dan klik LOGIN.



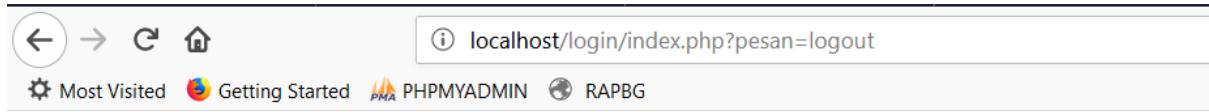
## Halaman Admin

Selamat datang, admin! anda telah login. ←

LOGOUT ←

contoh membuat login admin dengan php dan mysqli

login pun berhasil. kita coba klik pada link logout, maka halaman akan di alihkan kembali ke halaman login sambil menampilkan pesan notifikasi bahwa telah berhasil logout.



## Login - www.malasngoding.com

Anda telah berhasil logout

Username :

Password :

membuat logout dengan php dan mysqli

selesai. sampai di sini kita telah selesai membuat login dengan PHP dan MySQLi.

# Membuat Login Dengan PHP dan MySQLi Menggunakan MD5

Sekarang teman-teman sedang berada di tutorial Membuat Login Dengan PHP dan MySQLi Menggunakan MD5. sebenarnya tutorial cara membuat login sudah pernah di bahas berulang kali. seperti misalnya sebelumnya kita telah membahas tentang cara membuat login dengan php dan mysqli, membuat login dengan php dan mysql, sampai membuat login dengan php dan mysqli menggunakan enkripsi md5 pada password.

Tapi tutorial kali ini agak sedikit berbeda. di tutorial ini kita akan belajar membuat login dengan php dan mysqli. tapi menggunakan enkripsi md5 untuk keamanan dan kerahasiaan password pengguna yang tersimpan di database.

Ada banyak kelebihan jika kita menerapkan enkripsi md5 pada password pengguna sistem yang kita buat.

DALAM TUTORIAL MEMBUAT LOGIN DENGAN PHP DAN MYSQLI MENGGUNAKAN MD5 INI TIDAK DIJELASKAN LAGI SECARA DETAIL TENTANG KESELURUHAN FUNGSI. HANYA FOKUS PADA PEMBAHASAN PENGGUNAAN MD5 PADA PROSES LOGIN. KARENA TUTORIALINI MELANJUTKAN DARI TUTORIAL SEBELUMNYA. SILAHKAN TEMAN-TEMAN BACA PADA TUTORIAL YANG DISEBUTKAN BARUSAN UNTUK PENJELASAN KESELURUHANNYA.

Untuk teman-teman yang ingin mengikuti tutorial ini, silahkan terlebih dulu saya sarankan untuk mempelajari dan mengikuti tutorial sebelumnya yaitu tentang tutorial membuat login dengan PHP dan MySQLi. karena sebenarnya di sini kita akan mengikuti tutorial sebelumnya. hanya bedanya sekarang kita akan menerapkan enkripsi MD5 pada password pengguna sistem. agar lebih aman dan terjaga kerahasiaannya.

## Persiapan

Untuk persiapannya, silahkan kita buat dulu contoh database dan tabel admin di dalam database yang kita buat, pada tabel admin ini lah nantinya kita simpan username dan password admin dalam bentuk enkripsi md5.

Buat database baru dengan nama 'akademik'. kemudian buat tabel baru dengan nama 'admin', dengan isi nya : id, username, dan password.

id bertipe data int, sedangkan username dan password bertipe data varchar dan beri nilai 225. jangan lupa mencentang *auto\_increment* dan pilih primary pada id. karena kita akan menjadikan id sebagai *primary key*.

The screenshot shows the MySQL Workbench interface for creating a new table. The table name is set to 'admin'. The structure is defined with three columns: 'id' (INT, primary key, auto-increment), 'username' (VARCHAR(255)), and 'password' (VARCHAR(255)). The 'Storage Engine' is set to InnoDB. Red boxes highlight the table name, the 'id' column definition, and the primary key and auto-increment settings for the 'id' column.

membuat table admin

Setelah selesai membuat database akademik dan tabel admin. selanjutnya kita akan input data admin baru dengan password yang terenkripsi md5. perhatikan contoh berikut.

The screenshot shows the MySQL Workbench interface for inserting data into the 'admin' table. The data is being inserted into three columns: 'id' (value: 1), 'username' (value: 'admin'), and 'password' (value: 'MD5'). The 'Go' button at the bottom right is highlighted with a red arrow, indicating where to click to execute the insert query. Red boxes highlight the 'username' and 'password' fields.

membuat login dengan php dan mysqli menggunakan md5

Pada password kita isi saja passwordnya dengan 'admin123'. dan jangan lupa memilih md5 pada kolumn function. dan klik go.

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: akademik
- Table: admin
- Query results: SELECT \* FROM `admin`
- Number of rows: 1
- Row data:

	id	username	password
<input type="checkbox"/>	2	admin	0192023a7bbd73250516f069df18b500
- Action buttons: Edit, Copy, Delete, Export

login enkripsi password md5 php

seperti yang teman-teman lihat pada kolom merah di atas. data password yang tadinya kita input 'admin123'. sekarang telah terenkripsi dengan md5. jadi password bisa aman dari orang-orang yang melihat database kita.

Sampai di sini persiapan kita telah selesai membuat database dan tabel admin.

## Membuat Login Dengan PHP dan MySQL Menggunakan MD5

Buat sebuah folder baru dengan nama terserah teman-teman. untuk nama project kita. di sini saya membuat folder baru dengan nama "loginmd5" di 'C://xampp/htdocs/' .

Buat file baru di dalam folder "loginmd5" dengan nama koneksi.php.

koneksi.php

```
<?php
$koneksi = mysqli_connect("localhost", "root", "", "akademik");

// Check connection
if (mysqli_connect_errno()) {
    echo "Koneksi database gagal : " . mysqli_connect_error();
```

```
}
```

```
?>
```

Kemudian buat sebuah file baru lagi dengan nama index.php. pada file index.php ini kita akan membuat halaman loginnya.

### index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Login Dengan PHP dan MySQLi Menggunakan MD5 - WWW.MALASNGODING.COM</title>
</head>
<body>
    <h2>Login MD5 - www.malasngoding.com</h2>
    <br/>
    <!-- cek pesan notifikasi -->
    <?php
        if(isset($_GET['pesan'])) {
            if($_GET['pesan'] == "gagal") {
                echo "Login gagal! username dan password salah!";
            } else if($_GET['pesan'] == "logout") {
                echo "Anda telah berhasil logout";
            } else if($_GET['pesan'] == "belum_login") {
                echo "Anda harus login untuk mengakses halaman admin";
            }
        }
    ?>
    <br/>
    <br/>
    <form method="post" action="cek login.php">
        <table>
            <tr>
                <td>Username</td>
                <td>:</td>
                <td><input type="text" name="username" placeholder="Masukkan username"></td>
            </tr>
            <tr>
                <td>Password</td>
                <td>:</td>
                <td><input type="password" name="password" placeholder="Masukkan password"></td>
            </tr>
            <tr>
                <td></td>
                <td></td>
                <td><input type="submit" value="LOGIN"></td>
            </tr>
        </table>
    </form>
```

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

di sini tidak lagi di jelaskan secara detail. karena sudah di jelaskan pada tutorial sebelumnya.

karena di sini file yang akan memproses data yang di kirim dari form ini adalah cek\_login.php,

```
<form method="post" action="cek_login.php">
```

maka kita buat sebuah file baru lagi dengan nama cek\_login.php.

cek\_login.php

```
<?php
// mengaktifkan session php
session_start();

// menghubungkan dengan koneksi
include 'koneksi.php';

// menangkap data yang dikirim dari form
$username = $_POST['username'];
$password = md5($_POST['password']);

// menyeleksi data admin dengan username dan password yang sesuai
$data = mysqli_query($koneksi,"select * from admin where username='$username'
and password='$password'");

// menghitung jumlah data yang ditemukan
$cek = mysqli_num_rows($data);

if($cek > 0){
    $_SESSION['username'] = $username;
    $_SESSION['status'] = "login";
    header("location:admin/index.php");
} else{
    header("location:index.php?pesan=gagal");
}
?>
```

Nah perhatikan di sini, di sini kita menggunakan fungsi md5() untuk mengubah data password yang dikirim dari form ke dalam bentuk md5.

```
$password = md5($_POST['password']);
```

beda seperti tutorial sebelumnya yang tidak menggunakan md5.

baru kemudian data password yang sudah di ubah ke format md5 ini kita cocok kan dengan password yang ada pada database.

```
$data = mysqli_query($koneksi,"select * from admin where username='$username' and password='$password'");
```

tidak ada perbedaan dengan tutorial membuat login sebelumnya. cuma di sini kita menambahkan fungsi md5.

oke selanjutnya buat sebuah folder dengan nama "admin". karena saat login berhasil akan di alihkan ke direktori "admin". pada folder admin tersebut kita akan membuat semua halaman adminnya dalam folder admin ini.

buat sebuah file baru lagi dengan nama index.php dalam folder "admin".

admin/index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Membuat Login Dengan PHP dan MySQLi Menggunakan MD5 - WWW.MALASNGODING.COM</title>
</head>
<body>
    <h2>Halaman Admin</h2>

    <br/>

    <!-- cek apakah sudah login -->
    <?php
        session_start();
        if($_SESSION['status']!='login'){
            header("location:../index.php?pesan=belum_login");
        }
    ?>

    <h4>Selamat datang, <?php echo $_SESSION['username']; ?>! anda telah
    login.</h4>

    <br/>
    <br/>

    <a href="logout.php">LOGOUT</a>
```

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

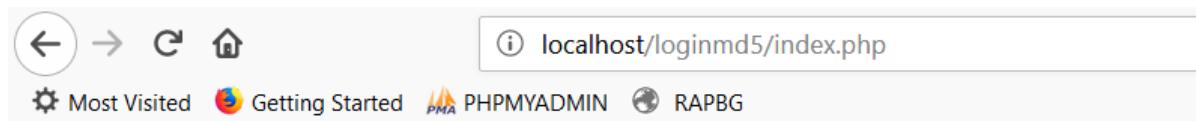
buat sebuah file baru lagi dengan nama logout.php dalam folder “admin”.

admin/logout.php

```
<?php  
// mengaktifkan session  
session_start();  
  
// menghapus semua session  
session_destroy();  
  
// mengalihkan halaman sambil mengirim pesan logout  
header("location:../index.php?pesan=logout");  
?>
```

Saat nya menjalankan proses login.

jalankan project pada browser dengan alamat <http://localhost/loginmd5>



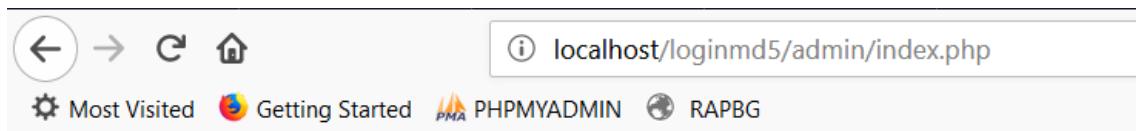
## Login MD5 - [www.malasngoding.com](http://www.malasngoding.com)

Username :

Password :

membuat login dengan php dan mysqli menggunakan md5

isi username : admin, dan password : admin123. klik login.



## Halaman Admin

Selamat datang, admin! anda telah login.

[LOGOUT](#)

sistem login berhasil

nah login pun berhasil.

Oke sekian dulu tutorial Membuat Login Dengan PHP dan MySQLi Menggunakan MD5.

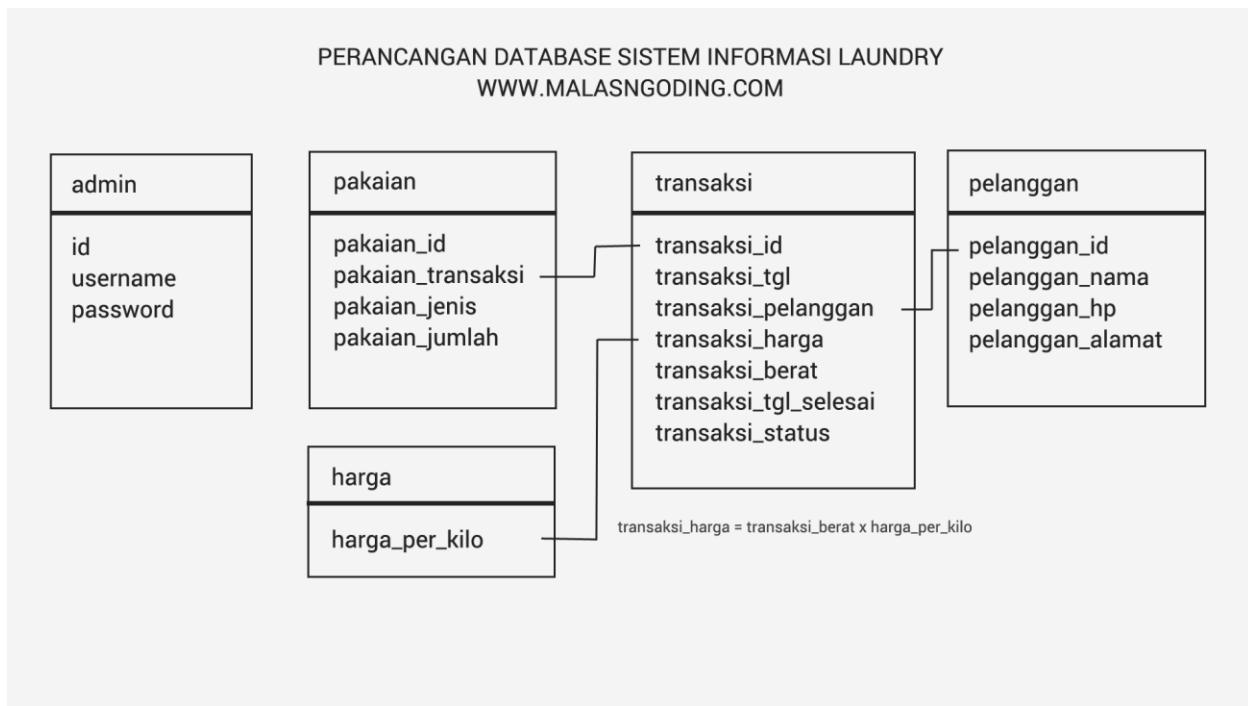
# **BELAJAR PHP TINGKAT LANJUTAN – MEMBUAT SISTEM INFORMASI LAUNDRY**

# Persiapan dan Penjelasan

Sebelum lebih jauh masuk ke proses pembuatan sistem informasi laundry, ada baiknya saya jelaskan dulu tentang sistem infomasi laundry yang akan kita buat.

Yang harus kita lakukan untuk membuat sebuah sistem infomasi adalah kita harus tau dulu alur aplikasi yang akan kita buat. Berikut adalah perancangan database yang telah saya siapkan.

Silahkan perhatikan pada gambar berikut



Seperti yang bisa teman-teman lihat pada gambar perancangan database sistem informasi laundry di atas. Gambar di atas merupakan gambaran perancangan database yang akan kita buat untuk sistem informasi laundry ini. Ada 5 tabel yang kita buruhkan, yaitu :

- admin : pada tabel ini kita akan menyimpan data username dan password admin.
- Pakaian : pada tabel ini kita menyimpan data pakaian atau data cucian si pelanggan, misalnya celana jeans 1, baju kemeja 2 dan lain-lain.
- Transaksi : pada tabel ini kita menyimpan data transaksi/invoice laundry, seperti tanggal transaksi, pelanggan yang melakukan transaksi, harga, berat,

tanggal cucian selesai dan bisa di ambil oleh pelanggan, serta status transaksi tersebut.

- Pelanggan : pada tabel ini kita menyimpan data pelanggan, misalnya nama, nomor hp dan alamat si pelanggan.
- Harga : pada tabel harga ini kita akan menyimpan pengaturan harga yang akan diterapkan untuk harga per kilo laundry atau cucian.

Untuk relasinya juga bisa teman-teman lihat pada gambar di atas, sudah di tandai dengan garis setiap relasi database atau tabel yang datanya berhubungan.

Selanjutnya, di sini saya menggunakan text editor sublime text editor. Teman-teman bisa bebas memilih ingin menggunakan text editor apa, misal notepad++, atom, dan lain-lain.

## Membuat Project Aplikasi PHP Baru

Sekarang kita akan membuat sebuah folder baru dengan nama 'laundry' dalam folder 'htdocs'. Karena di sini xampp saya terinsall di direktori C. maka folder 'laundry' nya saya buat di C://xampp/htdocs/.

Sampai disini pembuatan folder project aplikasi PHP kita telah selesai. Kita lanjutkan dulu ke persiapan yang lainnya seperti di bawah.

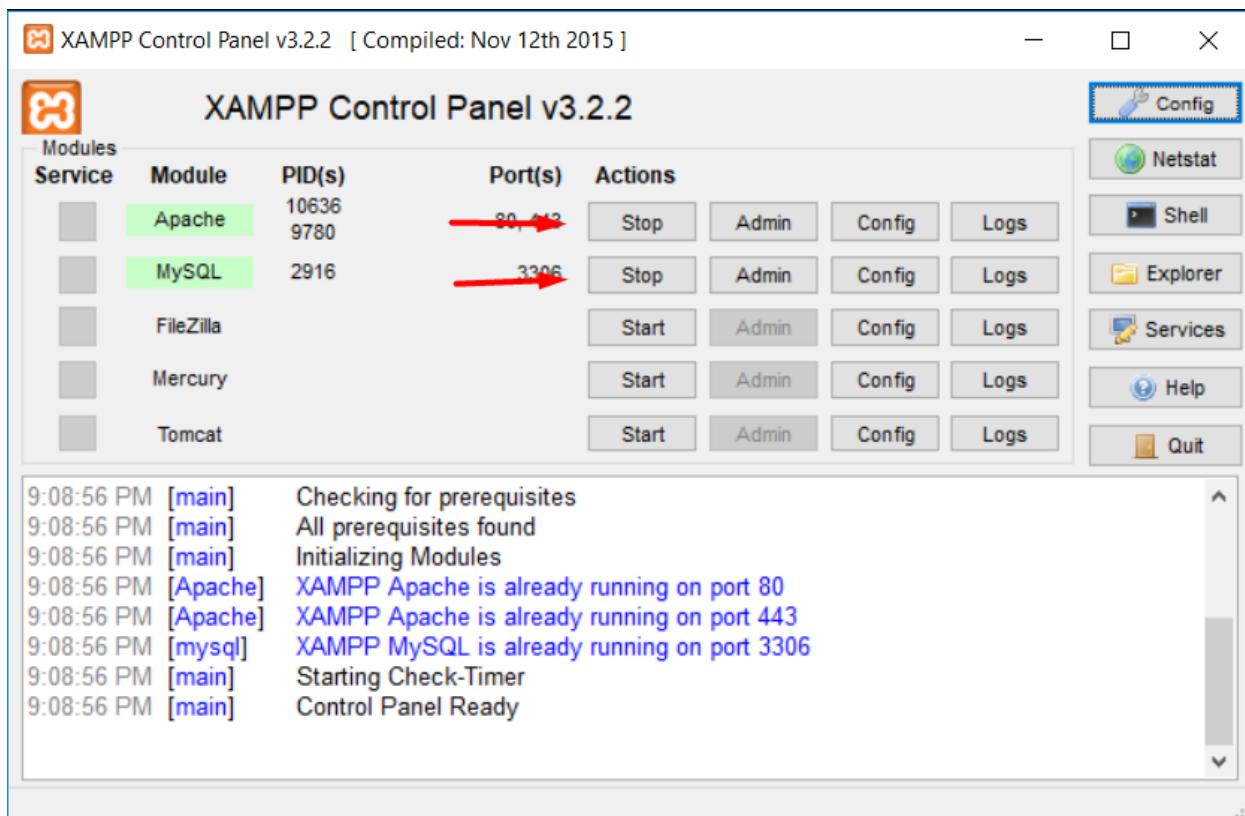
## Persiapan Database

Setiap aplikasi ataupun sistem informasi yang akan dibangun, pastinya membutuhkan sebuah database yang berguna untuk menampung data. Sebagai contohnya, pada sistem informasi laundry yang akan kita buat ini, semua data yang sudah dijelaskan pada sub judul sebelumnya, seperti data pelanggan, data admin, data transaksi dan data lainnya akan kita simpan di dalam sebuah database.

Nah, untuk sistem manajemen basis data nya, kita akan menggunakan MySQLi. Seperti yang telah dijelaskan sebelumnya, MySQLi adalah pengembangan dari MySQL. Yang tentunya MySQLi telah memiliki banyak kelebihan dibandingkan MySQL seperti yang pernah diuraikan di atas pada sub judul sebelumnya.

Karena sistem informasi laundry yang kita buat membutuhkan database, maka sekarang kita akan membuat sebuah database baru dengan nama 'laundry'.

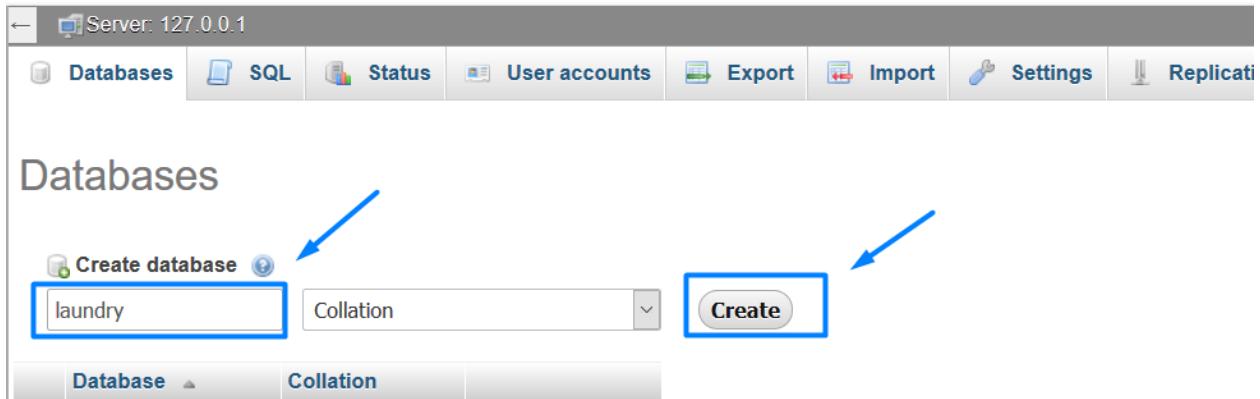
Caranya aktifkan dulu apache dan MySQL dengan menggunakan XAMPP. Caranya dengan klik tombol start pada Apache dan MySQL. Lihat gambar berikut.



#### Mengaktifkan Apache & MySQL

Selanjutnya buka PHPMyAdmin dengan cara mengakses "*localhost/phpmyadmin*" pada web browser.

Kemudian klik pada menu database, dan buat sebuah database baru dengan nama "laundry".



Database akan dibuat setelah anda klik pada tombol 'create'.

Sampai di sini kita telah selesai membuat database dengan nama "laundry".

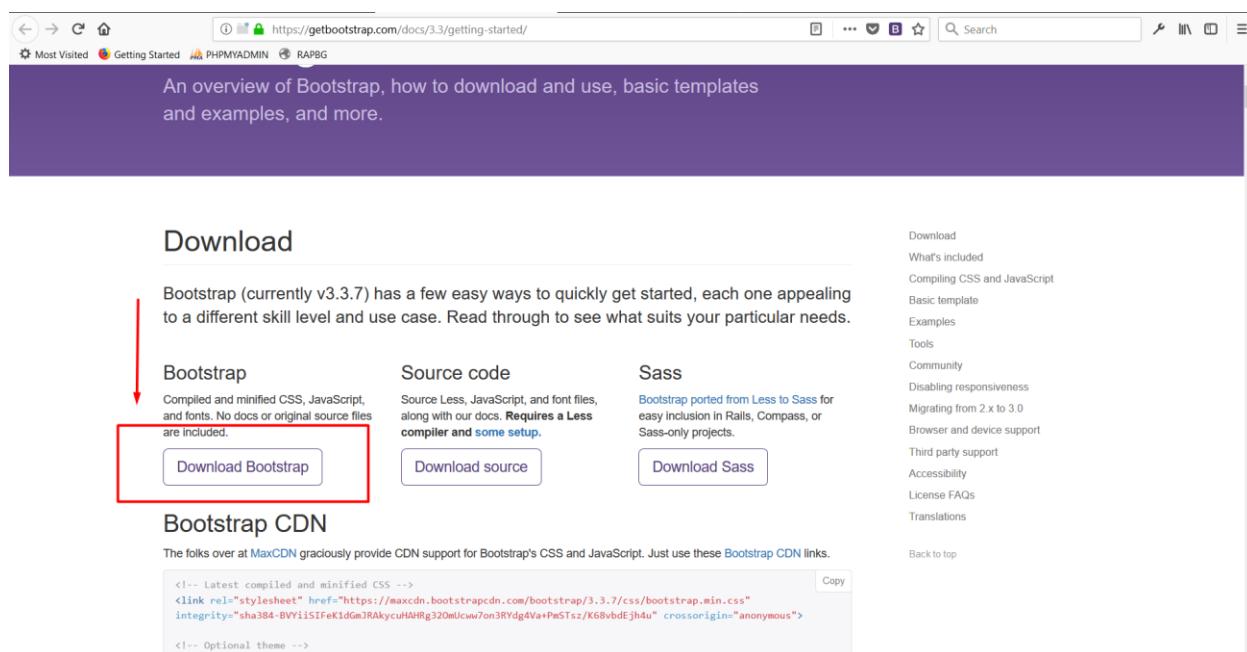
# Instalasi Bootstrap

Setelah membuat database laundry, kita tinggalkan sebentar masalah database, kita masuk ke tahap instalasi library CSS. Untuk pembuatan antarmuka atau user interface sistem informasi laundry yang akan kita buat ini, kita akan menggunakan bootstrap.

Pasti teman-teman telah familiar dengan nama yang namanya bootstrap. Di [www.malasngoding.com](http://www.malasngoding.com) sendiri telah ada pembahasan khusus tentang tutorial bootstrap yang telah lengkap dari part 1 sampai ke sekian, yang di mulai dari perkenalan bootstrap sampai membuat menu, membuat tabel dan lain-lainnya dengan mudah menggunakan bootstrap. Teman-teman bisa membacanya di [www.malasngoding.com/category/bootstrap](http://www.malasngoding.com/category/bootstrap).

## Instalasi Bootstrap

Bootstrap bisa langsung di download dari situs resminya, dengan cara mengakses alamat <https://getbootstrap.com/docs/3.3/getting-started/>.



The screenshot shows the official Bootstrap documentation page at <https://getbootstrap.com/docs/3.3/getting-started/>. The main content area has a purple header with the text "An overview of Bootstrap, how to download and use, basic templates and examples, and more.". Below this, there's a "Download" section with three main options:

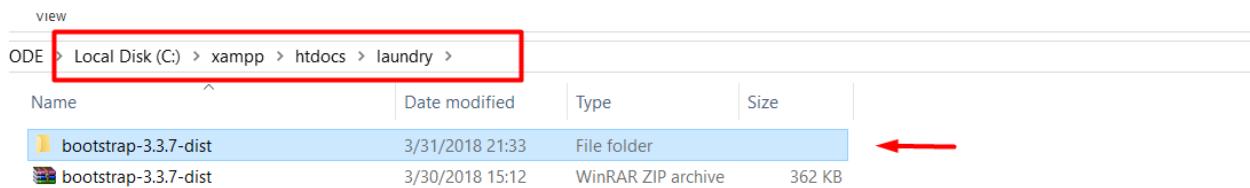
- Bootstrap**: Described as "Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included." It includes a red-bordered "Download Bootstrap" button.
- Source code**: Described as "Source Less, JavaScript, and font files, along with our docs. **Requires a Less compiler and some setup.**" It includes a "Download source" button.
- Sass**: Described as "Bootstrap ported from Less to Sass for easy inclusion in Rails, Compass, or Sass-only projects." It includes a "Download Sass" button.

To the right of the download section is a sidebar with links to other documentation sections like "Download", "What's included", "Compiling CSS and JavaScript", etc. At the bottom of the page, there's a "Bootstrap CDN" section with a snippet of CSS code for MaxCDN and a "Back to top" link.

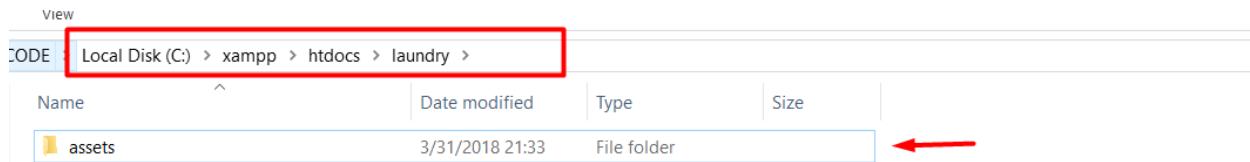
Atau teman-teman bisa menggunakan bootstrap yang telah saya sertakan dalam paket ebook ini yang berada dalam folder 'library'.

Saya anggap teman-teman telah memiliki bootstrap nya, sekarang copy dan paste kan bootstrap yang telah teman-teman download ke folder project kita. Yaitu ke folder 'laundry' yang telah kita buat pada C://xampp/htdocs/.

Extrak file bootstrap tersebut ke dalam folder 'laundry'.



Kemudian rename menjadi 'assets'. Dan hapus file .zip bootstrap nya.



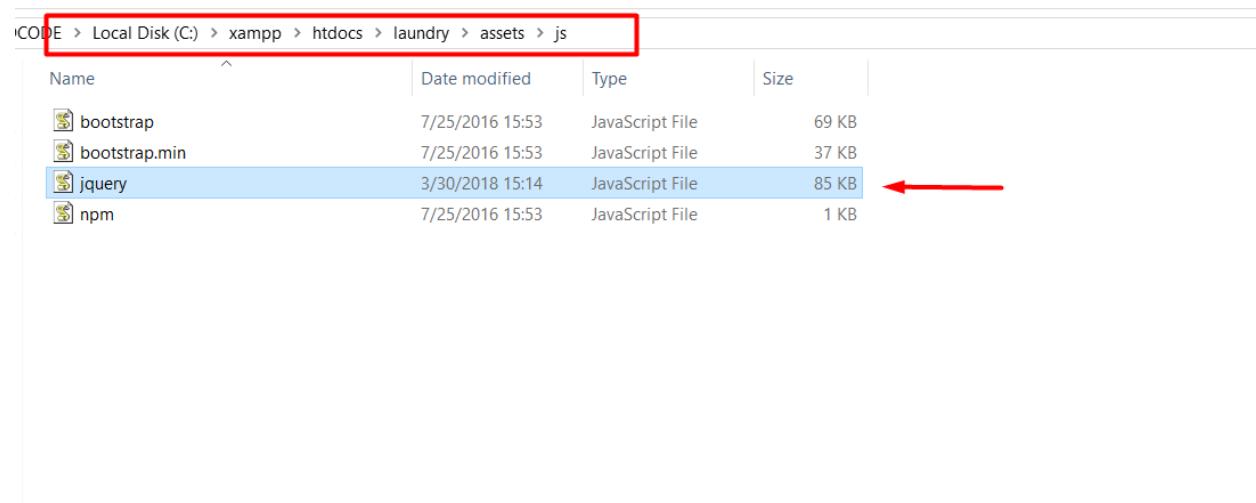
Oke selesai untuk tahap instalasi bootstrap. Kita masuk ke instalasi jquery.

## Instalasi Jquery

Bootstrap membutuhkan jquery untuk dapat berjalan dengan baik, karena bootstrap juga menggunakan jquery sebagai library javascript nya.

Oleh karena itu, sekarang kita akan mulai melakukan instalasi jquery. Caranya cukup mudah, silahkan teman-teman download langsung pada situs resmi jquery, yaitu di <https://jquery.com/>. Atau teman-teman bisa menggunakan jquery yang sudah saya sertakan pada paket ebook ini.

Copy dan paste kan file jquery.js nya ke dalam folder js yang berada dalam folder assets. Perhatikan gambar berikut.



Oke, selesai. Jika teman-teman mengikuti panduan instalasi bootstrap dan jquery ini dengan benar, maka instalasi nya telah selesai. Cukup mudah bukan.

## Koneksi Database

Setelah selesai melakukan instalasi bootstap dan lain-lain. Untuk tahap paling awalnya juga kita harus membuat koneksi database. Agar project PHP kita ini bisa terhubung dengan database 'laundry' yang telah kita buat sebelumnya.

Caranya sangat mudah. Buat sebuah file baru dengan nama *koneksi.php* pada folder *laundry*.

### koneksi.php

```
<?php  
  
// nama host, Username, password dan nama database  
$koneksi = mysqli_connect("localhost","root","","laundry");  
  
// Periksa Koneksi  
if (mysqli_connect_errno()){  
    echo "Koneksi database gagal : " . mysqli_connect_error();  
}  
  
?>
```

Oke, koneksi database selesai kita buat. Kita lanjutkan dengan membuat login untuk admin.

# Membuat Login Admin

Pada halaman utama aplikasi atau sistem informasi laundry yang akan kita buat ini kita akan membuat halaman login untuk admin. Jadi pada saat sistem informasi laundry yang kita buat ini di akses, maka yang pertama muncul adalah halaman login admin. Jika admin telah berhasil login, baru kemudian admin dapat mengakses fitur yang lainnya.

Pada sub judul membuat login admin ini tidak lagi dijelaskan secara detail tentang cara membuat login. Karena sudah dijelaskan pada sub judul sebelumnya. Pada sub judul ini hanya akan dijelaskan tahap-tahap membuat login untuk admin sesuai dengan kebutuhan kita di sistem informasi laundry yang akan kita buat ini.

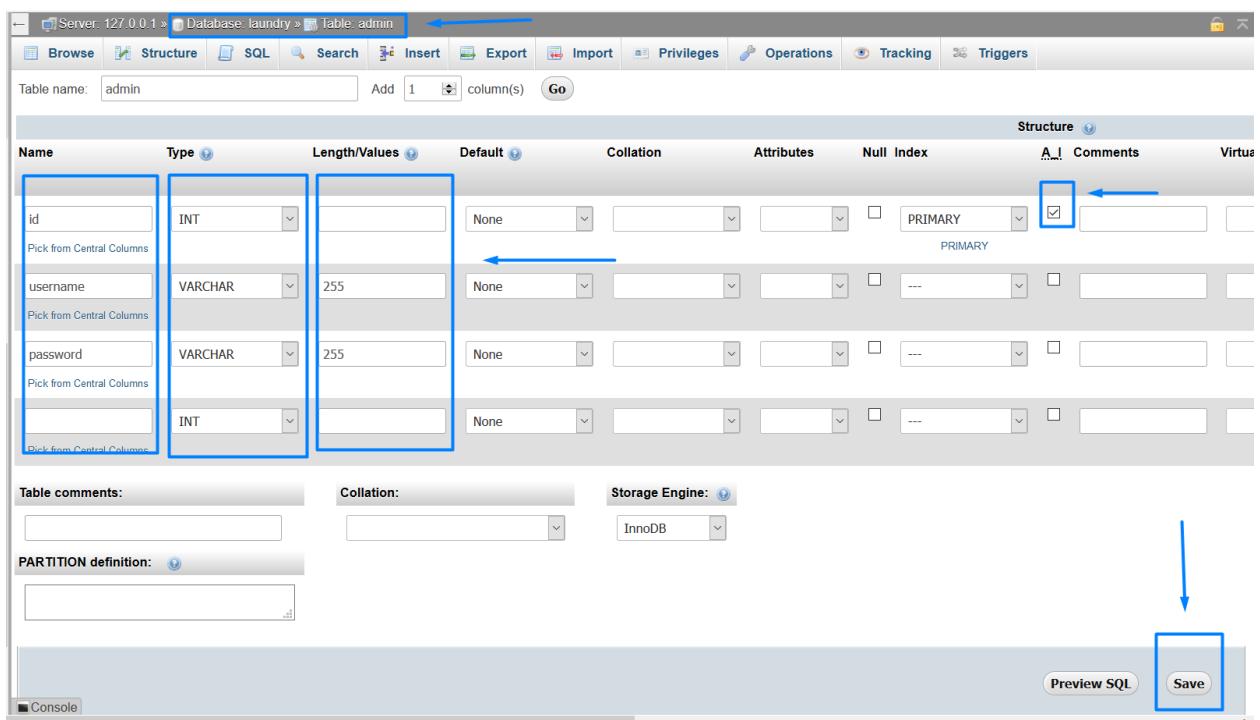
## Membuat Tabel Admin

Data admin sistem informasi laundry ini akan kita simpan pada tabel 'admin' dalam database 'laundry'.

Oleh sebab itu kita akan membuat sebuah tabel dengan nama 'admin'. Caranya silahkan pilih database laundry, kemudian pada form create table, tulis nama table yang ingin dibuat, tulis admin. Kemudian klik go.



Dengan kolumn id, username, dan password. Id ber tipe data int, sedangkan username password bertipe data varchar dengan nilai 255. Dan jangan lupa centang pada auto-increment seperti pada gambar di bawah.



Dan klik save.

Sampai di sini kita telah selesai membuat tabel admin. Selanjutnya kita akan menginput data admin sebagai contoh. Sekaligus untuk membuat akun admin yang akan digunakan untuk login pada pembahasan selanjutnya.

Klik pada menu insert, kemudian isikan usernamenya dengan 'admin'. Dan passwordnya 'admin123'. Jangan lupa untuk memilih fungsi md5 pada kolumn function.

Karena kita akan membuat login yang menggunakan enkripsi md5. Untuk penjelasan cara membuat login dengan php dan mysqli menggunakan md5 juga telah di bahas pada [www.malasngoding.com](http://www.malasngoding.com) dan pada sub judul sebelumnya dalam ebook ini.

Server: 127.0.0.1 » Database: laundry » Table: admin

Column	Type	Function	Null	Value
id	int(11)			1
username	varchar(255)			admin
password	varchar(255)	MD5		admin123

**Go** ←

Klik go, dan selesai.

+ Options

	id	username	password
<input type="checkbox"/>	1	admin	0192023a7bbd73250516f069df18b500

← ↑  Check all With selected:  Edit  Copy  Delete  Edit  Copy  Delete  Export ←

Data admin pun selesai kita input dan password yang tadinya kita input dengan 'admin123' pun berhasil di generate secara otomatis menjadi enkripsi md5.

## Membuat Halaman Login

Seperti yang telah dijelaskan sebelumnya, halaman login yang akan kita buat ini akan muncul pada saat pertama kali sistem informasi laundry ini di akses atau di jalankan.

Oleh karena itu kita akan membuat halaman login nya pada file index.php. karena file index.php adalah file php yang pertama kali di jalankan di dalam suatu folder.

Langsung saja buat sebuah file baru dengan nama index.php pada folder laundry.

D CODE > Local Disk (C:) > xampp > htdocs > laundry

Name	Date modified	Type	Size
assets	3/31/2018 21:33	File folder	
index	3/30/2018 15:58	PHP File	2 KB
koneksi	3/30/2018 15:59	PHP File	1 KB

Dan ketik atau copas syntax berikut.

### Index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Sistem Informasi Laundry Malas Ngoding - www.malasngoding.com</title>

    <link rel="stylesheet" type="text/css" href="assets/css/bootstrap.css">
    <script type="text/javascript" src="assets/js/jquery.js"></script>
    <script type="text/javascript" src="assets/js/bootstrap.js"></script>

</head>
<body style="background: #f0f0f0">
    <br/>
    <br/>
    <br/>
    <br/>

    <center>
        <h2>SISTEM INFORMASI LAUNDRY <br/> www.malasngoding.com</h2>
    </center>

    <br/>
    <br/>
    <br/>
    <br/>

    <div class="container">
        <div class="col-md-4 col-md-offset-4">
```

```

<?php
if(isset($_GET['pesan'])){
    if($_GET['pesan'] == "gagal"){
        echo "<div class='alert alert-danger'>Login gagal! username dan password salah!</div>";
    }else if($_GET['pesan'] == "logout"){
        echo "<div class='alert alert-info'>Anda telah berhasil logout</div>";
    }else if($_GET['pesan'] == "belum_login"){
        echo "<div class='alert alert-danger'>Anda harus login untuk mengakses halaman admin</div>";
    }
}
?>

<form action="login.php" method="post">
    <div class="panel">
        <br/>
        <div class="panel-body">
            <div class="form-group">
                <label>Username</label>
                <input type="text" class="form-control" name="username">
            </div>
            <div class="form-group">
                <label>Password</label>
                <input type="password" class="form-control" name="password">
            </div>
            <input type="submit" class="btn btn-primary" value="Log In">
        </div>
        <br/>
    </div>
</form>
</div>
</body>
</html>

```

Saya rasa sudah banyak coding yang teman-teman tau kegunaan nya dari coding di atas. Karena sudah berulang-ulang kali di jelaskan pada sub judul sebelumnya.

Saya jelaskan sedikit di sini, perhatikan pada

```

<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.css">
<script type="text/javascript" src="assets/js/jquery.js"></script>
<script type="text/javascript" src="assets/js/bootstrap.js"></script>

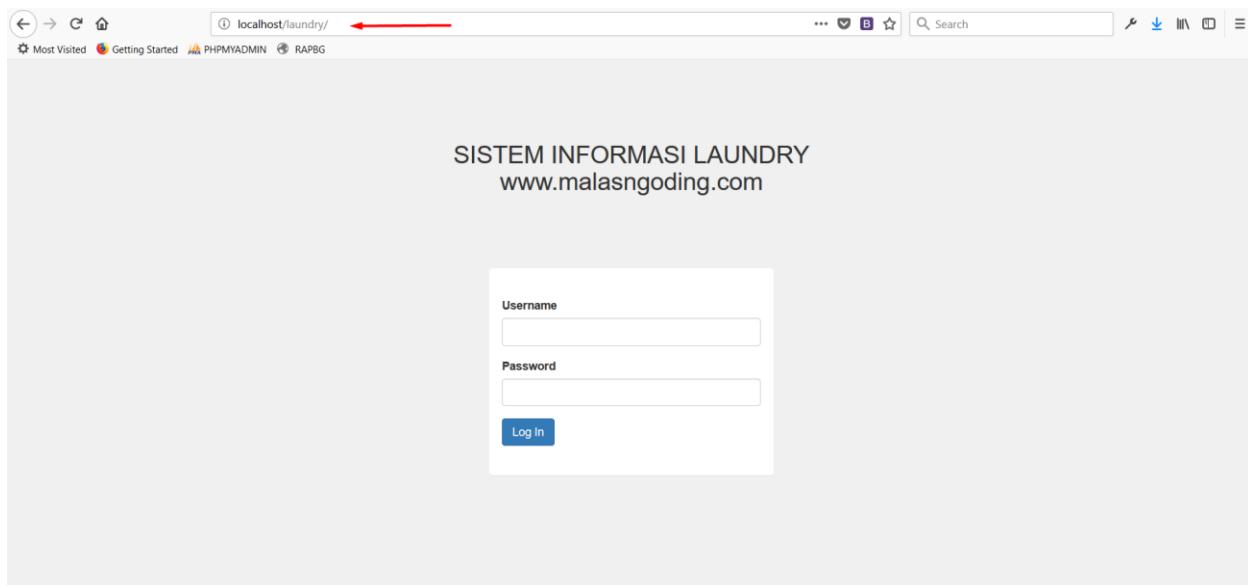
```

Syntax di atas merupakan syntax yang kita buat untuk menghubungkan dengan bootstrap.css, javascript.js, dan file jquery yang telah kita copy ke folder laundry ini pada sub judul sebelumnya.

Selanjutnya pada bagian bawah, kita membuat form seperti pada tutorial membuat login sebelumnya yang telah kita bahas juga.

Hanya di sini bedanya pasti teman-teman menemukan class dengan nama panel, panel-body, form-group dan lain-lain. Itu adalah class dari library bootstrap yang telah kita pelajari juga di [www.malasngoding.com/category/bootstrap](http://www.malasngoding.com/category/bootstrap).

Sekarang coba kita lihat bagaimana hasilnya, silahkan akses project kita dengan cara mengakses "localhost/laundry" pada browser.



## Membuat Aksi Login Admin

Perhatikan pada tag form di atas tadi.

```
<form action="login.php" method="post">
```

Pada form login ini kita telah menentukan bahwa form ini akan di eksekusi atau di proses pada file login.php. dengan menggunakan method post.

Kemudian buat sebuah file baru lagi dengan nama login.php dalam folder laundry.

## login.php

```
<?php
// mengaktifkan session php
session_start();

// menghubungkan dengan koneksi
include 'koneksi.php';

// menangkap data yang dikirim dari form
$username = $_POST['username'];
$password = md5($_POST['password']);
// fungsi md5 di atas untuk enkripsi kedalam bentuk md5

// menyeleksi data admin dengan username dan password yang sesuai
$data = mysqli_query($koneksi,"select * from admin where username='$username'
and password='$password'");

// menghitung jumlah data yang ditemukan
$cek = mysqli_num_rows($data);

// cek jika username dan password yang di input di temukan, buat session dan
// alihkan halaman ke halaman admin(folder admin),
// jika tidak, alihkan kembali ke halaman depan sambil mengirim pesan gagal
if($cek > 0){
    $_SESSION['username'] = $username;
    $_SESSION['status'] = "login";
    header("location:admin/index.php");
} else{
    header("location:index.php?pesan=gagal");
}
?>
```

Perhatikan lagi, karena pada file login.php ini kita akan berhubungan dengan database, maka kita haru menghubungkan dulu file ini dengan file koneksi.php. caranya dengan meng-include file koneksi.php seperti pada syntax di atas.

```
// menghubungkan dengan koneksi
include 'koneksi.php';
```

Selanjutnya seperti pada penjelasan pada sub judul membuat login dengan php dan mysqli yang telah dibahas sebelumnya.

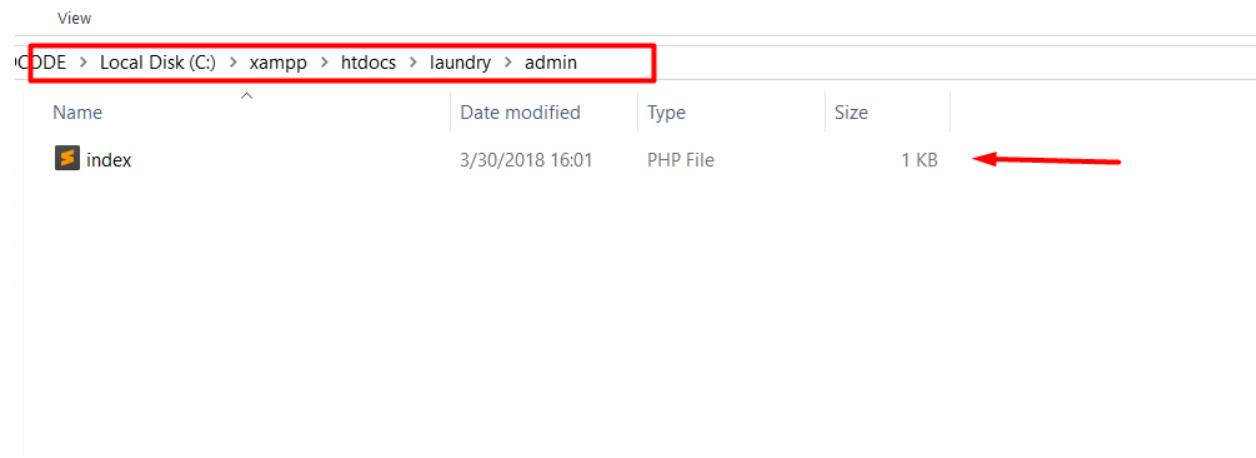
Oh iya, pada pengecekan jika login berhasil, maka halaman akan kita alihkan ke file index.php yang berada dalam folder admin.

```
header("location:admin/index.php");
```

Karena kita akan membuat halaman admin sistem informasi laundry ini di dalam folder admin. Jadi file-file php untuk admin semuanya kita buat dalam folder admin. Agar source code sistem informasi laundry yang kita buat lebih bersih dan teratur.

Buat sebuah folder baru dengan nama 'admin' dalam folder laundry.

Kemudian buat sebuah file baru lagi dengan nama 'index.php' di dalam folder admin. Sehingga hasilnya seperti berikut.



## Index.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Sistem Informasi Laundry Malas Ngoding - WWW.MALASNGODING.COM</title>
</head>
<body>
    <h2>Halaman Admin</h2>
    <br/>
    <!-- cek apakah sudah login -->
    <?php
        session_start();
        if($_SESSION['status']!='login'){
            header("location:../index.php?pesan=belum_login");
        }
    ?>

    <h4>Selamat datang, <?php echo $_SESSION['username']; ?>! anda telah login.</h4>
    <p>selamat datang di halaman admin. halaman admin masih kosong. yuk buat halaman admin nya dulu biar kece.</p>
```

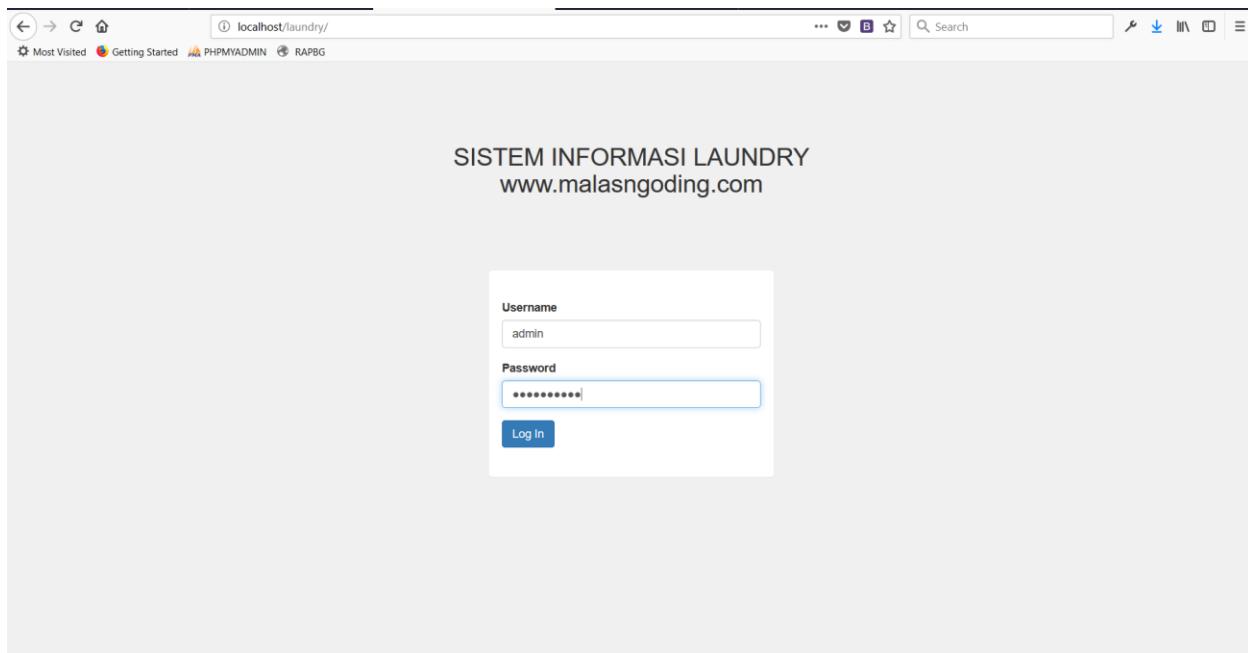
```
<br/>
<br/>

<a href="logout.php">LOGOUT</a>

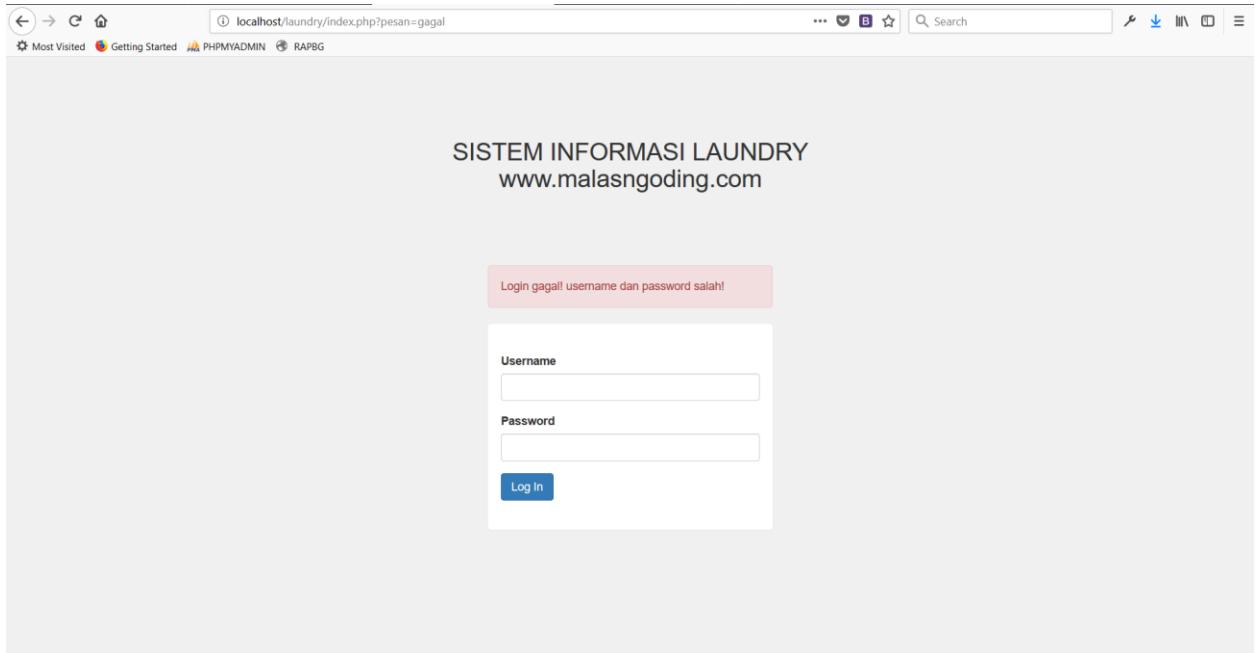
</body>
</html>
```

Sekarang coba jalankan pada browser. Akses dengan alamat “localhost/laundry”.

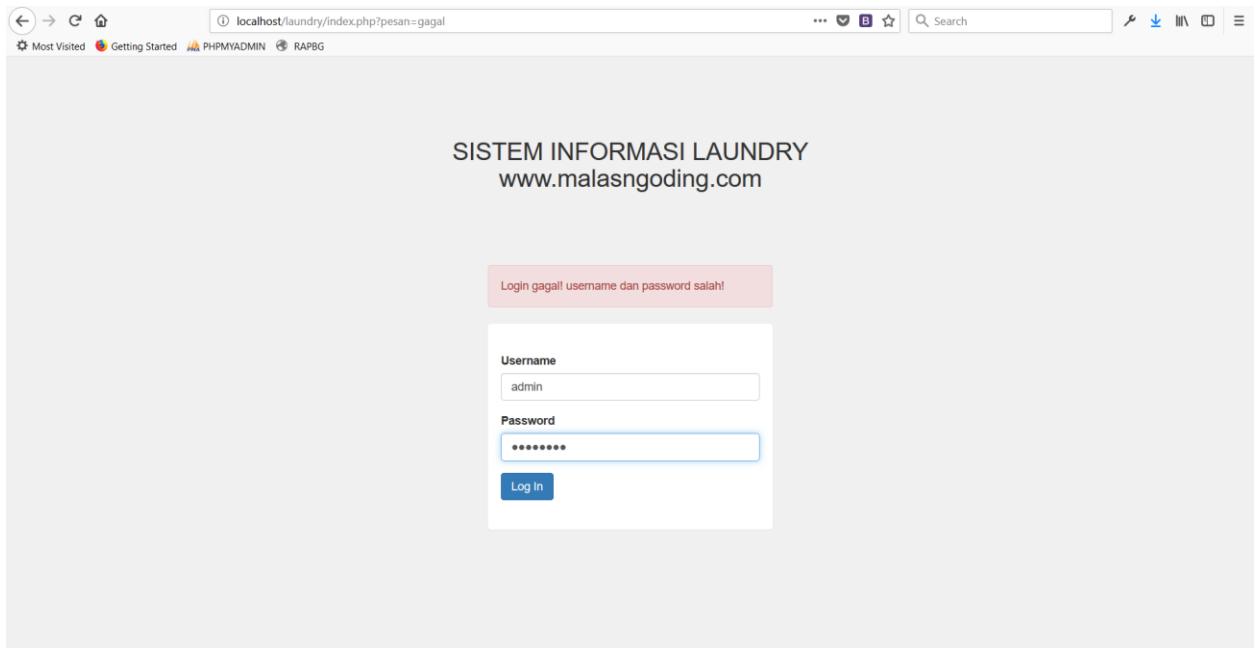
Pertama kita coba dulu dengan menginput username dan password yang salah. Untuk mengecek apakah sudah berjalan dengan seharusnya. Di sini saya mencoba memasukkan username “admin” dan password “admin12345”.



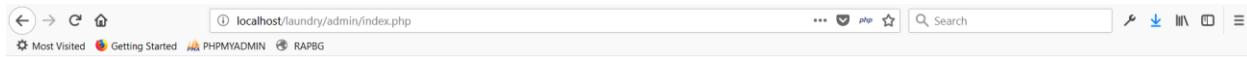
Dan klik log in. maka akan muncul pesan notifikasi bahwa login gagal.



Sekarang coba masukkan username dan password yang benar. Seperti username dan password yang kita input ke table admin pada sub judul sebelumnya. Yaitu username “admin” dan password “admin123”.



Kemudian klik log in.



## Halaman Admin

Selamat datang, admin! anda telah login.

selamat datang di halaman admin. halaman admin masih kosong. yuk buat halaman admin nya dulu biar kece.

[LOGOUT](#)

Nah maka kita telah berhasil login. Tapi tampilan loginnya masih sederhana. Untuk tampilan dashboard adminnya akan kita buat pada sub judul selanjutnya.

## Membuat Logout Admin

Setelah sebelumnya kita belajar cara membuat login dengan php dan mysqli, pada sub judul ini kita akan belajar cara membuat logout admin dengan php. Caranya sangat mudah, bahkan pada sub judul sebelumnya kita telah belajar cara membuat logoutnya juga pada sub judul tutorial membuat login dengan php dan mysqli.

Perhatikan pada halaman admin yang telah kita buat di atas, di sana terdapat link **LOGOUT** yang telah kita buat, dan saat di klik halaman akan di alihkan ke file **logout.php**.

```
<a href="logout.php">LOGOUT</a>
```

Oleh karena itu sekarang kita akan membuat file **logout.php**. untuk membuat proses penghapusan semua session dan mengalihkan kembali admin ke halaman login.

Buat file **logout.php** di dalam folder admin, berdampingan dengan file **index.php**. seperti gambar di bawah.

View				
CODE > Local Disk (C:) > xampp > htdocs > laundry > admin				
Name	Date modified	Type	Size	
index	3/30/2018 16:01	PHP File	1 KB	
logout	4/2/2018 17:21	PHP File	1 KB	←

## logout.php

```
<?php
// mengaktifkan session
session_start();

// menghapus semua session yang ada
session_destroy();

// mengalihkan halaman kembali ke halaman login sambil mengirimkan pesan
// logout melalui url
header("location:../index.php?pesan=logout");
?>
```

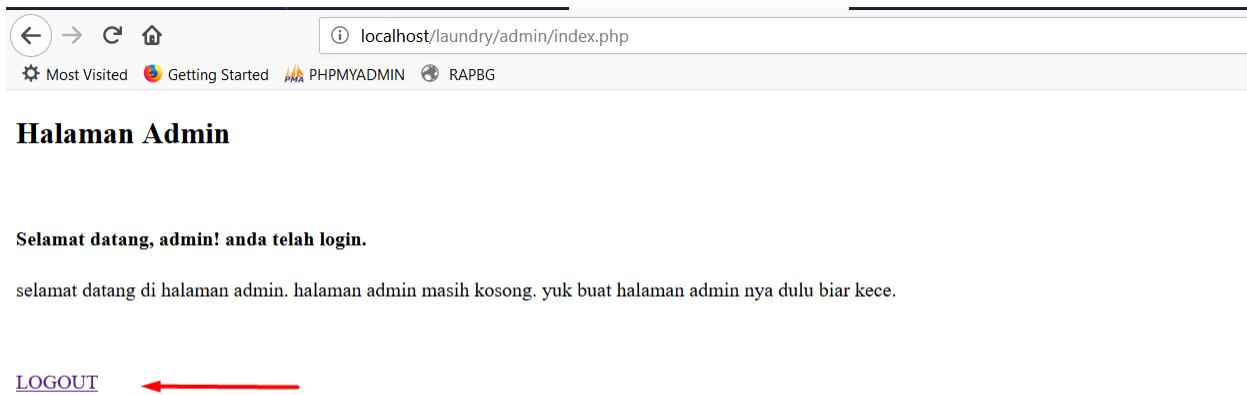
Jika kita ingin menggunakan fungsi session dalam sebuah file PHP, kita wajib menggunakan session\_start() pada bagian paling atas, agar session PHP di aktifkan saat file php tersebut di jalankan.

```
// mengaktifkan session
session_start();
```

Kemudian kita menggunakan fungsi session\_destroy() untuk menghapus semua session yang ada, dan yang terakhir setelah session di hapus, kita alihkan halaman ke halaman login.

```
header("location:../index.php?pesan=logout");
```

Oke selesai. Sekarang kita coba untuk klik link logout. Maka session akan dihapus dan halaman akan dialihkan kembali ke halaman login.



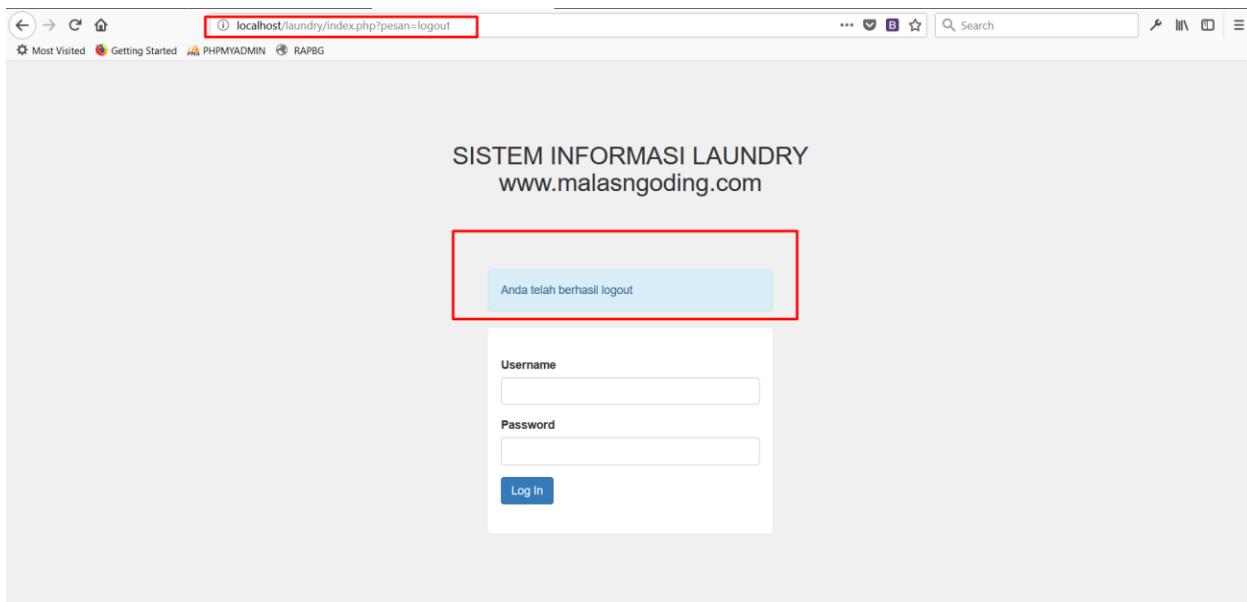
The screenshot shows a browser window with the following details:

- Address bar: localhost/laundry/admin/index.php
- Toolbar icons: Back, Forward, Stop, Home, Refresh.
- Links in the toolbar: Most Visited, Getting Started, PHPMYADMIN, RAPBG.
- Page title: Halaman Admin
- Content:

Selamat datang, admin! anda telah login.  
selamat datang di halaman admin. halaman admin masih kosong. yuk buat halaman admin nya dulu biar kece.

[LOGOUT](#) ←

Maka halaman akan dialihkan kembali ke halaman login.



The screenshot shows a browser window with the following details:

- Address bar: localhost/laundry/index.php?pesan=logout
- Toolbar icons: Back, Forward, Stop, Home, Refresh.
- Links in the toolbar: Most Visited, Getting Started, PHPMYADMIN, RAPBG.
- Page title: SISTEM INFORMASI LAUNDRY  
www.malasngoding.com
- Content:

Anda telah berhasil logout

Username

Password

Dan logout pun berhasil kita buat. Jika teman-teman masih bingung dengan proses membuat login sistem informasi laundry ini, teman-teman bisa membaca tutorial nya kembali pada pembahasan sebelumnya tentang membuat login dengan php dan mysqli. Karena sudah di jelaskan secara detail, sampai penjelasan membuat pesan notifikasi di atas.

## Membuat Halaman Admin (Halaman Dinamis)

Setelah selesai membuat login dan logout, maka selanjutnya kita buat halaman dashboard admin untuk admin sistem informasi laundry ini dengan menggunakan bootstrap. Agar lebih keren.

Halaman dashboard yang sudah kita buat sebelumnya (index.php) masih sangat sederhana dan belum terintegrasi dengan bootstrap. Sekarang kita akan membuat halaman dashboard admin dengan menggunakan bootstrap.

Adapun komponen bootstrap yang kita gunakan di antaranya adalah navbar, glyphicon, panel, alert dan masih ada beberapa lagi. Kembali saya ingatkan untuk teman-teman yang ingin belajar memahami penggunaan bootstrap secara lebih detail, teman-teman bisa membaca tutorialnya di malasngoding.com dengan url berikut : <https://www.malasngoding.com/category/bootstrap>. atau teman-teman bisa juga mempelajarinya melalui panduan dari situs resmi bootstrap sendiri. Yaitu di url : <https://getbootstrap.com/docs/3.3/components/>.

Pada halaman dashboard admin ini, kita akan menerapkan halaman dinamis. Silahkan teman-teman baca lagi tutorial pada sub judul sebelumnya tentang cara membuat halaman web dinamis. Agar tidak bingung dalam mengikuti panduan ini. Seperti yang sudah di jelaskan sebelumnya, pada pembuatan sistem informasi laundry ini kita tidak lagi membahas secara mendetail kecuali hal-hal yang belum pernah di jelaskan pada tutorial dasar pada sub judul sebelumnya yang telah di jelaskan secara detail.

Langsung saja kita mulai. Silahkan teman-teman buka file index.php pada folder admin. Sekalian buat 2 buah file php lagi. Yaitu file header.php. dan footer.php.

Berikut ini adalah syntax halaman dashboard admin yang telah jadi.

```
<!DOCTYPE html>
<html>
<head>
    <title>Sistem Informasi Laundry Malas Ngoding - WWW.MALASNGODING.COM</title>
    <link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.css">
    <script type="text/javascript" src="../assets/js/jquery.js"></script>
    <script type="text/javascript" src="../assets/js/bootstrap.js"></script>
```

```

</head>
<body style="background: #f0f0f0">
    <!-- cek apakah sudah login -->
    <?php
        session_start();
        if($_SESSION['status']!="login"){
            header("location:../index.php?pesan=belum_login");
        }
    ?>

    <!-- menu navigasi -->
    <nav class="navbar navbar-inverse" style="border-radius: 0px">
        <div class="container-fluid">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="index.php">LAUNDRY
Malas Ngoding</a>
            </div>
            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="index.php"><i class="glyphicon glyphicon-home"></i> Dashboard</a></li>
                    <li><a href="pelanggan.php"><i class="glyphicon glyphicon-user"></i> Pelanggan</a></li>
                    <li><a href="transaksi.php"><i class="glyphicon glyphicon-random"></i> Transaksi</a></li>
                    <li><a href="laporan.php"><i class="glyphicon glyphicon-list-alt"></i> Laporan</a></li>
                    <li class="dropdown">
                        <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false"><i class="glyphicon glyphicon-wrench"></i> Pengaturan <span class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="harga.php"><i class="glyphicon glyphicon-usd"></i> Pengaturan Harga</a></li>
                            <li><a href="ganti_password.php"><i class="glyphicon glyphicon-lock"></i> Ganti Password</a></li>
                        </ul>
                    </li>
                    <li><a href="logout.php"><i class="glyphicon glyphicon-log-out"></i> Log Out</a></li>
                </ul>
                <ul class="nav navbar-nav navbar-right">
                    <li><p class="navbar-text">Halo, <b><?php echo $_SESSION['username']; ?></b> !</p></li>

```

```

        </ul>
    </div>
</div>
<!-- akhir menu navigasi -->

<div class="container">
    <div class="alert alert-info text-center">
        <h4 style="margin-bottom: 0px"><b>Selamat datang!</b> di
sistem informasi laundry malas ngoding.</h4>
    </div>

    <div class="panel">
        <div class="panel-heading">
            <h4>Dashboard</h4>
        </div>
        <div class="panel-body">
            Sistem Informasi Laundry Malas Ngoding
        </div>
    </div>
</div>

</body>
</html>

```

Sekarang kita pecah menjadi 3 buah file. Bagian header dashboard kita simpan dalam file header.php. bagian body kita simpan pada file index.php. dan bagian footer kita simpan pada file footer.php. semua file yang akan kita buat tetap dalam folder admin.

Dari syntax di atas, bagian menu navigasi akan kita simpan ke file header.php. buat sebuah file header.php. dan tulis syntax berikut.

### Header.php

```

<!DOCTYPE html>
<html>
<head>
    <title>Sistem Informasi Laundry Malas Ngoding -<br/>WWW.MALASNGODING.COM</title>

    <link rel="stylesheet" type="text/css"
href="../assets/css/bootstrap.css">
    <script type="text/javascript" src="../assets/js/jquery.js"></script>
    <script type="text/javascript"
src="../assets/js/bootstrap.js"></script>

</head>

```

```

<body style="background: #f0f0f0">
    <!-- cek apakah sudah login -->
    <?php
        session_start();
        if($_SESSION['status'] != "login") {
            header("location:../index.php?pesan=belum_login");
        }
    ?>

    <!-- menu navigasi -->
    <nav class="navbar navbar-inverse" style="border-radius: 0px">
        <div class="container-fluid">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="index.php">LAUNDRY
Malas Ngoding</a>
            </div>
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                <li class="active"><a href="index.php"><i class="glyphicon glyphicon-home"></i> Dashboard</a></li>
                <li><a href="pelanggan.php"><i class="glyphicon glyphicon-user"></i> Pelanggan</a></li>

                <li><a href="transaksi.php"><i class="glyphicon glyphicon-random"></i> Transaksi</a></li>
                <li><a href="laporan.php"><i class="glyphicon glyphicon-list-alt"></i> Laporan</a></li>
                    <li class="dropdown">
                        <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false"><i class="glyphicon glyphicon-wrench"></i> Pengaturan <span class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="harga.php"><i class="glyphicon glyphicon-usd"></i> Pengaturan Harga</a></li>
                            <li><a href="ganti_password.php"><i class="glyphicon glyphicon-lock"></i> Ganti Password</a></li>
                        </ul>
                    </li>
                    <li><a href="logout.php"><i class="glyphicon glyphicon-log-out"></i> Log Out</a></li>
                </ul>
                <ul class="nav navbar-nav navbar-right">
                    <li><p class="navbar-text">Halo, <b><?php echo $_SESSION['username']; ?></b> !</p></li>
                </ul>
            </ul>
        </div>
    </nav>

```

```
        </div>
    </div>
</nav>
<!-- akhir menu navigasi --&gt;</pre>
```

Kemudian buka file index.php, dan tulis syntax berikut.

### Index.php

```
<?php include 'header.php'; ?>

<div class="container">
    <div class="alert alert-info text-center">
        <h4 style="margin-bottom: 0px"><b>Selamat datang!</b> di sistem
informasi laundry malas ngoding.</h4>
    </div>

    <div class="panel">
        <div class="panel-heading">
            <h4>Dashboard</h4>
        </div>
        <div class="panel-body">
            Sistem Informasi Laundry Malas Ngoding
        </div>
    </div>
</div>

<?php include 'footer.php'; ?>
```

Kemudian sisanya kita simpan pada file footer.php. jadi buat sebuah file baru dengan nama footer.php

### Footer.php

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

Akan saya jelaskan sedikit. Bagian header dan bagian footer kita include dengan menggunakan function “ include ‘’ ” seperti pada file index.php, yaitu

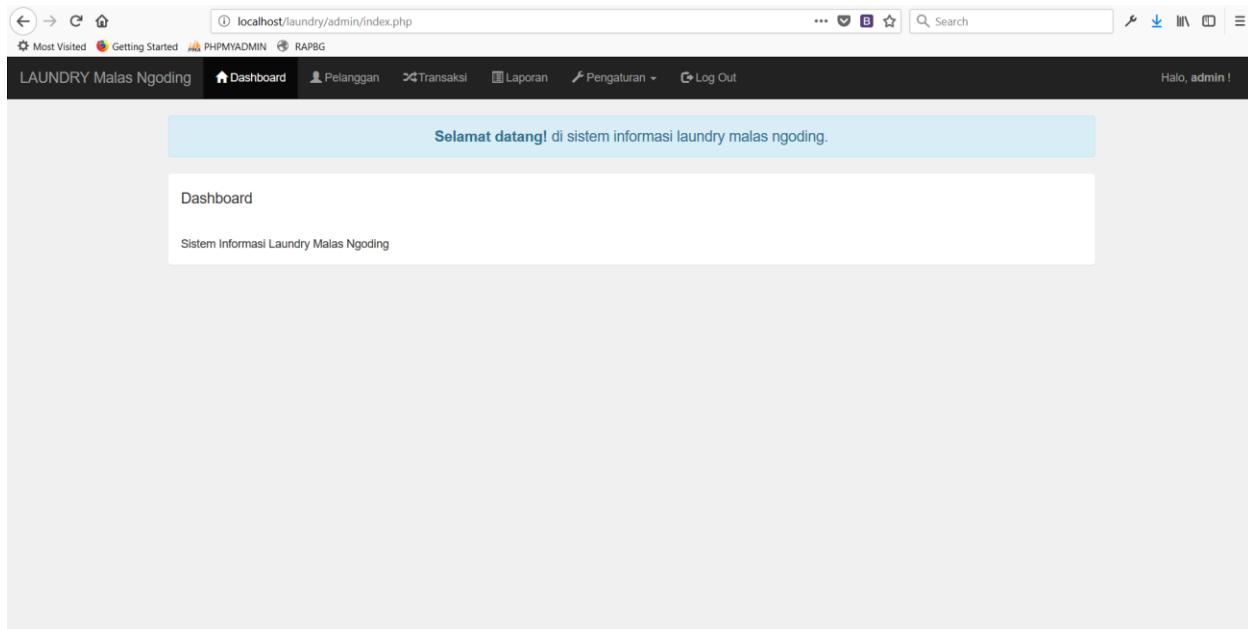
```
<?php include 'header.php'; ?>
```

Dan

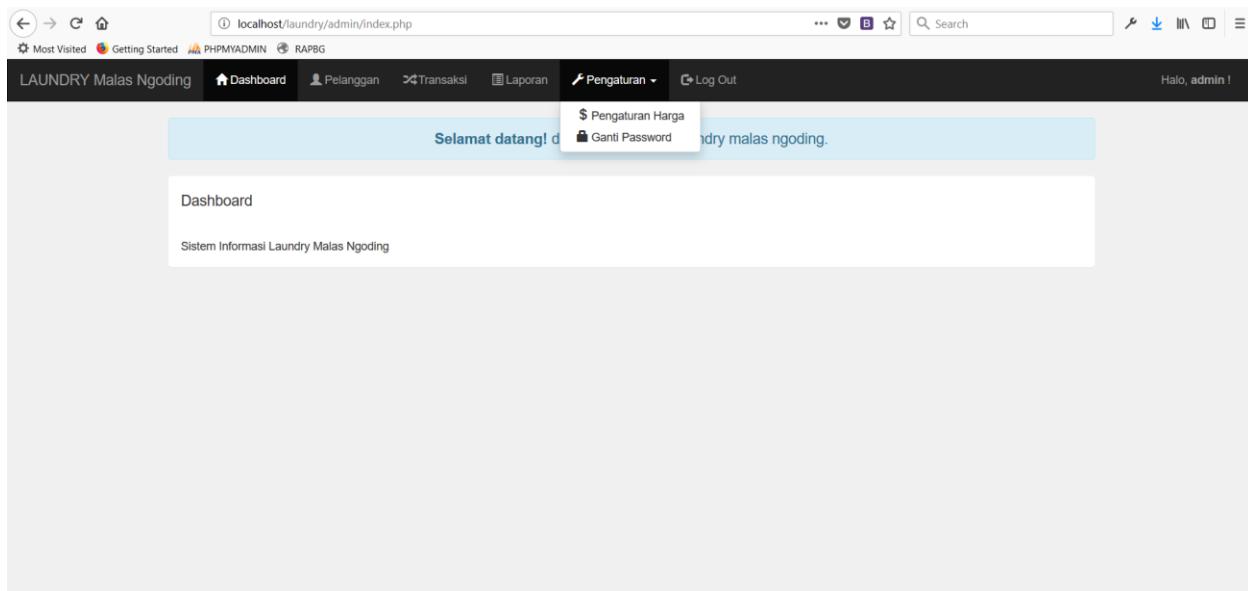
```
<?php include footer.php'; ?>
```

Sampai di sini selesai lah halaman dashboard admin kita untuk sementara. Sekarang coba refresh halaman admin kita.

Dan hasilnya akan seperti berikut.



Dan sub menunya juga.



Nah sudah jadi semuanya. Di sini saya juga telah membuat menu-menu navigasi yang akan kita alihkan ke halaman-halaman yang kita butuhkan. Seperti pelanggan, transaksi, laporan dan lain-lain.

Sampai di sini kita telah selesai membuat halaman dinamis pada halaman dashboard admin.

## Membuat Ganti Password Admin

Setelah segala persiapan halaman dashboard admin selesai, kita akan melanjutkan dengan membuat ganti password untuk admin.

Perhatikan pada menu ganti password di atas.

```
<li><a href="ganti_password.php"><i class="glyphicon glyphicon-lock"></i>  
Ganti Password</a></li>
```

Pada menu ganti password telah kita tetapkan link url tujuannya. Yaitu mengarah ke file ganti\_password.php. karena form untuk pergantian password admin akan kita buat pada file ganti\_password.php.

Buat sebuah file baru dalam folder admin, dengan nama ganti\_password.php.

### Ganti\_password.php

```
<?php include 'header.php'; ?>

<div class="container">
    <br/>
    <br/>
    <br/>
    <div class="col-md-5 col-md-offset-3">

        <?php
        if(isset($_GET['pesan'])){
            if($_GET['pesan'] == "oke"){
                echo "<div class='alert alert-success'>Password telah
diganti!</div>";
            }
        }
        ?>

        <div class="panel">
            <div class="panel-heading">
                <h4>Ganti Password</h4>
```

```

        </div>
        <div class="panel-body">

            <form method="post" action="ganti_password_aksi.php">
                <div class="form-group">
                    <label>Masukkan Password Baru</label>
                    <input type="password" class="form-control" name="password_baru" placeholder="Masukkan Password Baru Anda ..." />
                </div>

                <br/>

                <input type="submit" class="btn btn-primary" value="Ganti Password" />
            </form>

        </div>
    </div>

<?php include 'footer.php'; ?>

```

Perhatikan pada tag `<form>` pada form ganti password di atas. Kita menetapkan action form ganti password ini ke file ganti\_password\_aksi.php

### Ganti\_password\_aksi.php

```
<form method="post" action="ganti_password_aksi.php">
```

Karena kita akan memproses data yang di kirim dari form ganti password ini di file ganti\_password\_aksi.php.

Buat sebuah file baru lagi dalam folder admin dengan nama ganti\_password\_aksi.php.

### Ganti\_password\_aksi.php

```

<?php

// menghubungkan dengan koneksi
include '../koneksi.php';

// menangkap data yang dikirim dari form
$password_baru = md5($_POST['password_baru']);
// fungsi md5 di atas untuk enkripsi kedalam bentuk md5

// mengupdate data password pada table admin

```

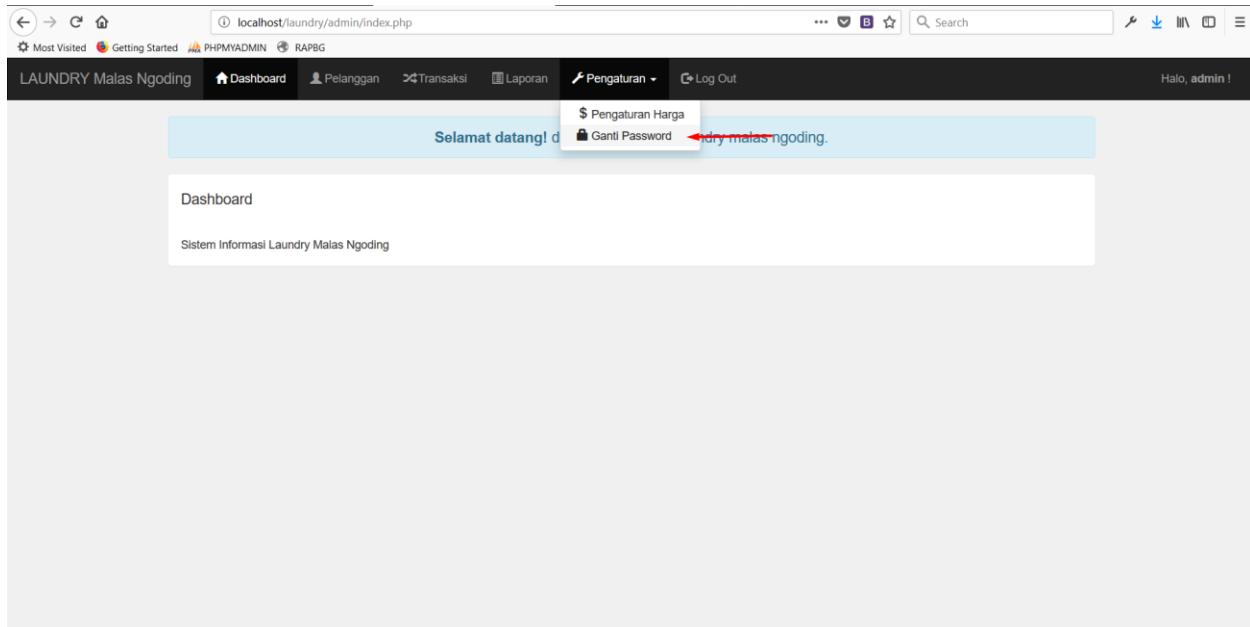
```
mysqli_query($koneksi,"update admin set password='".$password_baru"');

header("location:ganti_password.php?pesan=oke");

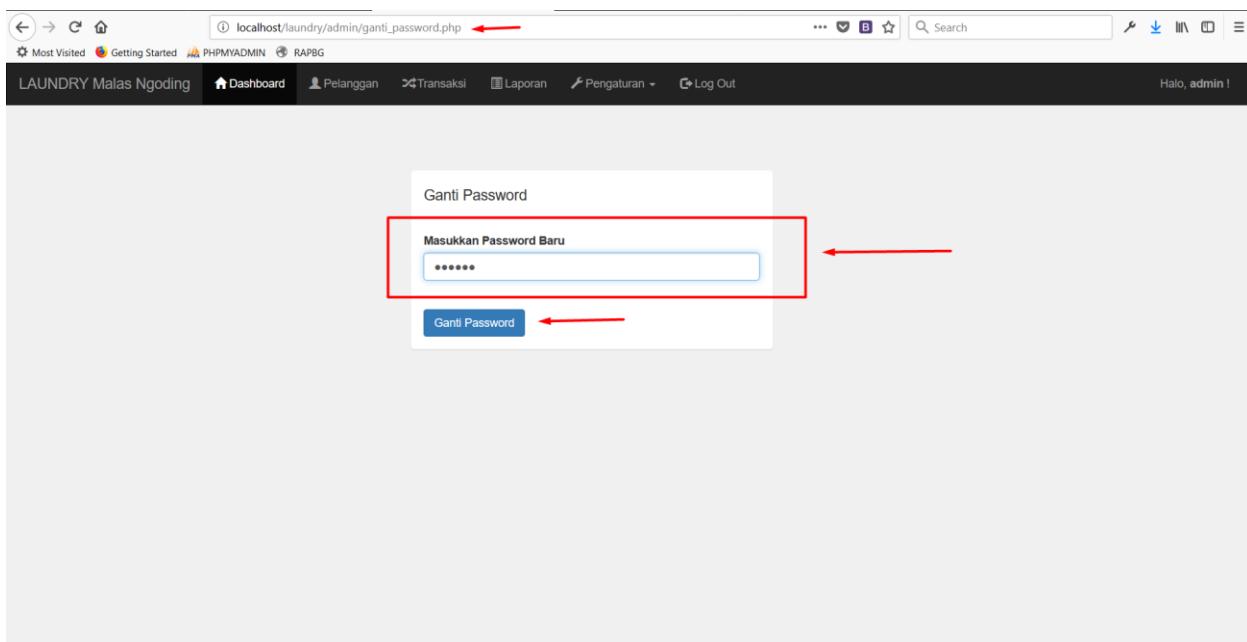
?>
```

Saya rasa teman-teman sudah paham dengan syntax di atas. Karena pada sub judul sebelumnya telah kita bahas berulang kali.

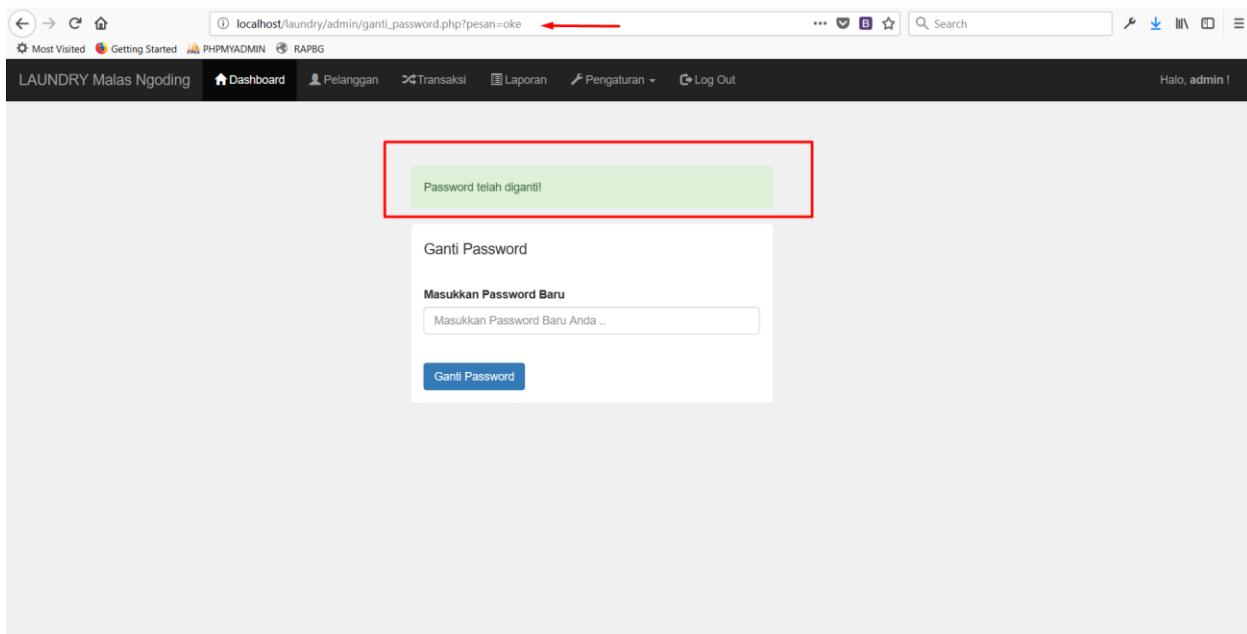
Sekarang coba kita jalankan pada browser. Klik pada menu ganti password. Dan masukkan password baru. Di sini saya memasukkan dengan password “tes123”.



Akan di alihkan ke halaman ganti password (ganti\_password.php).



Saya mencoba memasukkan password baru dengan “tes123”. Klik pada tombol ganti password.



Maka data password baru yang kita inputpun di proses dengan pertama menangkap data yang dikirim dari form ganti password kemudian di ubah ke enkripsi md5 menggunakan fungsi md5() di php.

```
// menangkap data yang dikirim dari form
$password_baru = md5($_POST['password_baru']);
```

Selanjutnya menjalankan query sql untuk mengupdate password pada table password dengan password yang baru.

```
// mengupdate data password pada table admin  
mysqli_query($koneksi,"update admin set password='$password_baru'");
```

Terakhir kita alihkan kembali halamannya ke ganti\_password.php. sambil mengirimkan pesan oke melalui url.

```
header("location:ganti_password.php?pesan=oke");
```

Sehingga bisa teman-teman lihat pada ganti\_password.php. di sana kita telah membuat pengecekan jika terdapat data pesan oke yang dikirim melalui url akan kita tampilkan pesan notifikasi jika password telah berhasil di ganti.

```
<?php  
if(isset($_GET['pesan'])) {  
    if($_GET['pesan'] == "oke") {  
        echo "<div class='alert alert-success'>Password telah  
diganti!</div>";  
    }  
}  
?>
```

Selesailah sudah tahap membuat ganti password untuk admin.

Oh ya, pada baris pertama, kita menghubungkan file php ini dengan file koneksi.php dengan syntax berikut.

```
include '../koneksi.php';
```

Kita menggunakan “ .. ” (titik dua kali) karena file koneksi.php tidak berada di dalam satu folder dengan file ganti\_password\_aksi.php. tetapi file koneksi.php berada di luar folder admin. Jadi istilahnya tanda “ .. ” di gunakan sebagai kembali ke direktori atau folder sebelumnya.

# Membuat CRUD Pelanggan Dengan PHP dan MySQLi

Seperti yang sudah dijelaskan sebelumnya, CRUD merupakan singkatan dari Create, Read, Update dan Delete. CRUD adalah istilah untuk proses input data ke database, menampilkan data dari database, mengupdate data di database dan menghapus data dari database.

Dan sesuai dengan perancangan database yang telah kita buat. Dalam sistem informasi laundry ini kita akan membuat CRUD data pelanggan. Agar bisa mengelola data pelanggan dari database menggunakan PHP.

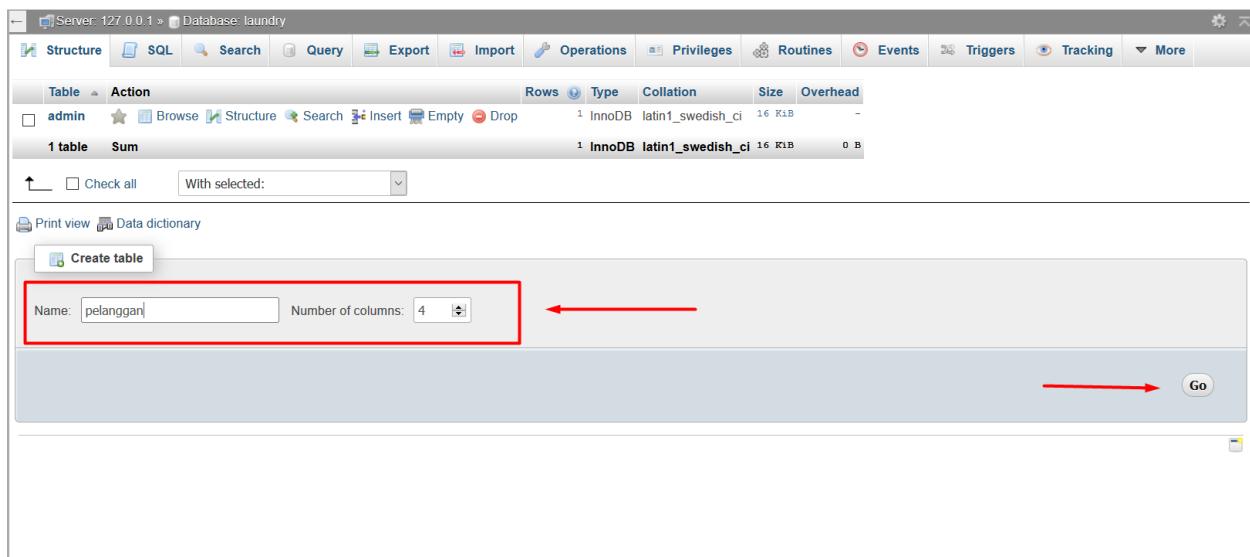
Untuk tutorial membuat CRUD php dan MySQLi sudah di jelaskan pada sub judul sebelumnya. Di sini kita hanya menerapkan kembali tutorial CRUD PHP dan MySQLi sebelumnya.

Langsung saja kita mulai untuk membuat CRUD Pelanggan.

## Membuat Tabel Pelanggan

Karena kita akan menyimpan data pelanggan pada database, jadi sekarang kita wajib membuat sebuah tabel baru dengan nama “pelanggan”.

Pasti teman-teman sudah tau cara membuat tabel di phpmyadmin jika teman-teman mengikuti tutorial cara membuat crud dengan php dan musqli pada sub judul sebelumnya.



Buat nama tabel dengan nama “pelanggan”. Dan klik go.

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	Comments	Virtua
pelanggan_id	INT		None				PRIMARY	<input checked="" type="checkbox"/>	
pelanggan_nama	VARCHAR	255	None				...	<input type="checkbox"/>	
pelanggan_hp	VARCHAR	20	None				...	<input type="checkbox"/>	
pelanggan_alamat	TEXT		None				...	<input type="checkbox"/>	

Kolumn yang akan kita buat di antaranya adalah pelanggan\_id, pelanggan\_nama, pelanggan\_hp, pelanggan\_alamat. Dan jangan lupa untuk mencentang auto\_increment pada pelanggan\_id seperti biasa. Pelanggan\_id ber tipe data int, pelanggan\_nama bertipe data varchar dan bernilai 255, pelanggan\_hp bertipe data varchar dan bernilai 20. terakhir pelanggaan\_alamat bertipe data text. Kemudian klik save.

Setelah tabel pelanggan selesai dibuat, sekarang kita coba untuk menginput beberapa data testing sebagai data pelanggan. Sekaligus untuk data saat kita mencoba menampilkan data pelanggan dari database.

Pilih tabel pelanggan, dan klik insert.

The screenshot shows the MySQL Workbench interface for the 'pelanggan' table. The table has four columns: 'pelanggan\_id' (int(11)), 'pelanggan\_nama' (varchar(255)), 'pelanggan\_hp' (varchar(20)), and 'pelanggan\_alamat' (text). The 'Information' panel displays space usage (16 KiB), row statistics (Format: Compact, Collation: latin1\_swedish\_ci), and creation details (Apr 03, 2018 at 07:45 PM).

Kemudian isikan beberapa data testing untuk data pelanggan.

The screenshot shows the 'Insert' screen for the 'pelanggan' table. Data is being entered into the fields: 'pelanggan\_nama' is set to 'Sumarto', 'pelanggan\_hp' is set to '08123536738', and 'pelanggan\_alamat' is set to 'Jl. senderawasih no.126 bandung'. A red box highlights the 'Go' button at the bottom right of the form.

Setelah teman-teman mengisi data pelanggannya. Klik pada tombol go.

Seperti yang teman-teman lihat pada gambar berikut, saya sudah menginput beberapa data pelanggan.

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** laundry
- Table:** pelanggan
- Query Result:** SELECT \* FROM `pelanggan`
- Rows:** 2 total rows shown.
- Data:**

	pelanggan_id	pelanggan_nama	pelanggan_hp	pelanggan_alamat
<input type="checkbox"/>	1	Sumanto	08123536738	jl. cenderawasih no.126 bandung
<input type="checkbox"/>	2	Sri Dayatun	08736363444	jl. tangkuban perahu no.23 Bandung
- Action Buttons:** Edit, Copy, Delete for each row.

Nah, sampai di sini kita telah selesai membuat tabel dengan nama pelanggan. Dan telah selesai menginput beberapa data pelanggan. Yaitu data pelanggan yang bernama sumanto dan sri dayatun.

## Menampilkan Data Pelanggan Dari Database

Coba perhatikan pada link menu pelanggan. Menu pelanggan akan mengarah pada file pelanggan.php. jadi sekarang kita buat sebuah file baru dengan nama pelanggan.php.

Buat sebuah file baru dengan nama "pelanggan.php" masih dalam folder admin.

### Pelanggan.php

```
<?php include 'header.php'; ?>

<div class="container">
    <div class="panel">
        <div class="panel-heading">
            <h4>Data Pelanggan</h4>
        </div>
        <div class="panel-body">

            <a href="pelanggan_tambah.php" class="btn btn-sm btn-info pull-right">Tambah</a>

            <br/>
            <br/>

        </div>
    </div>
</div>
```

```


| No | Nama | HP | Alamat | OPSI |
|----|------|----|--------|------|
|----|------|----|--------|------|


<?php
// koneksi database
include '../koneksi.php';

// mengambil data pelanggan dari database
$data = mysqli_query($koneksi,"select * from
pelanggan");
$no = 1;
// mengubah data ke array dan menampilkannya dengan
perulangan while
while($d=mysqli_fetch_array($data)){
    ?>
    <tr>
        <td><?php echo $no++; ?></td>
        <td><?php echo $d['pelanggan_nama']; ?>
    </td>
        <td><?php echo $d['pelanggan_hp']; ?>
    </td>
        <td><?php echo $d['pelanggan_alamat']; ?>
    </td>
        <a href="pelanggan_edit.php?id=<?php echo $d['pelanggan_id']; ?>" class="btn
btn-sm btn-info">Edit</a>
        <a href="pelanggan_hapus.php?id=<?php echo $d['pelanggan_id']; ?>" class="btn
btn-sm btn-danger">Hapus</a>
    </td>
    </tr>
    <?php
}
    ?>

```

Seperti yang teman-teman lihat pada syntax di atas, untuk cara menampilkan data dari database, sama saja caranya seperti yang pernah kita pelajari pada sub judul membuat CRUD dengan PHP dan MySQLi sebelumnya tentang cara menampilkan data dari database.

Hanya sedikit perbedaannya. Yaitu pada class-class yang terdapat seperti pada table dan lain-lain. Class tersebut adalah class-class dari bootstrap yang tinggal kita gunakan saja. Teman-teman bisa mempelajari bootstrap langsung pada situs resminya di <https://getbootstrap.com/docs/3.3/components/> atau di [www.malasngoding.com](http://www.malasngoding.com) juga telah telah di jelaskan tentang tutorial bootstrap dari dasar sampai selesai.

Klik menu pelanggan. Dan lihat hasilnya.

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>

Data pelanggan telah berhasil kita tampilkan.

## Menginput Data Pelanggan Ke Database

Pada tahap ini kita akan membuat form untuk penginputan data pelanggan. Untuk cara menginput data ke database menggunakan PHP dan MySQLi telah di bahas juga pada sub judul membuat crud dengan PHP dan MySQLi.

Silahkan perhatikan pada link tombol tambah pada halaman pelanggan.php di atas.

```
<a href="pelanggan_tambah.php" class="btn btn-sm btn-info pull-right">Tambah</a>
```

Pada saat tombol tambah ini di klik, maka halaman akan di alihkan ke pelanggan\_tambah.php.

Pada file pelanggan\_tambah.php ini lah akan kita buat form penginputan data pelanggan.

Buat sebuah file baru masih dalam folder admin dengan nama "pelanggan\_tambah.php".

### Pelanggan\_tambah.php

```
<?php include 'header.php'; ?>

<div class="container">
    <br/>
    <br/>
    <br/>
    <div class="col-md-5 col-md-offset-3">

        <div class="panel">
            <div class="panel-heading">
                <h4>Tambah Pelanggan Baru</h4>
            </div>
            <div class="panel-body">

                <form method="post" action="pelanggan_aksi.php">
                    <div class="form-group">
                        <label>Nama</label>
                        <input type="text" class="form-control" name="nama" placeholder="Masukkan nama ..">
                    </div>

                    <div class="form-group">
                        <label>HP</label>
                        <input type="number" class="form-control" name="hp" placeholder="Masukkan no.hp ..">
                    </div>

                    <div class="form-group">
                        <label>Alamat</label>
                        <input type="text" class="form-control" name="alamat" placeholder="Masukkan alamat ..">
                    </div>

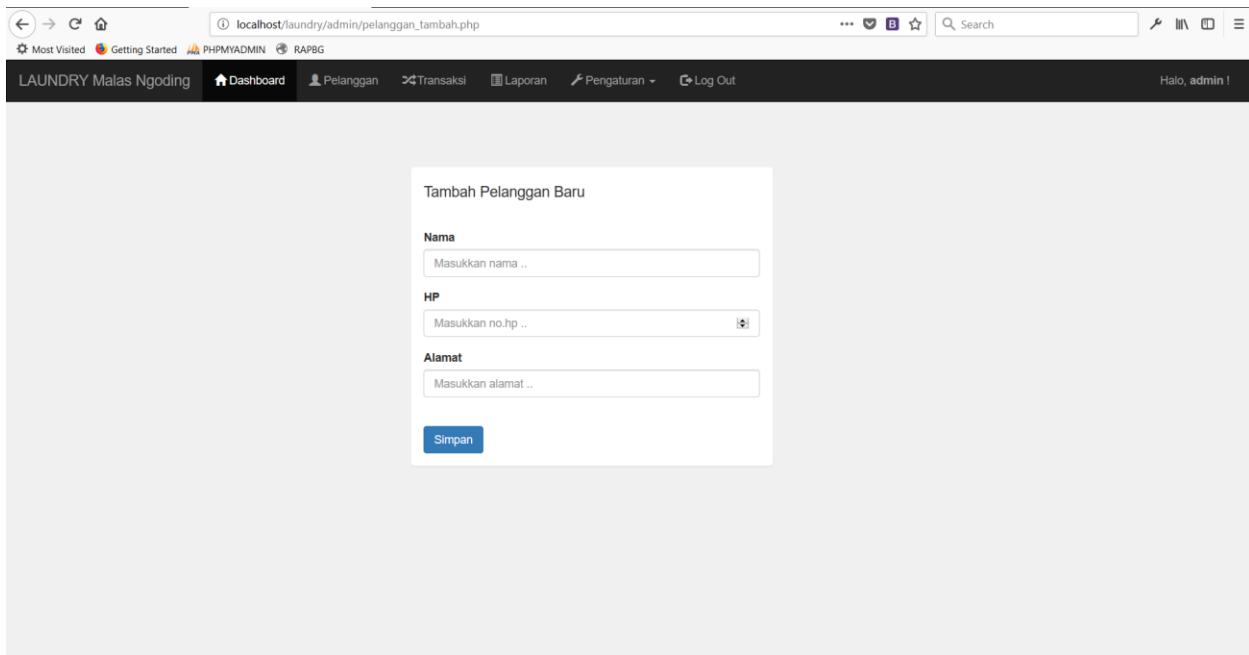
                    <br/>

                    <input type="submit" class="btn btn-primary" value="Simpan">
                </form>

            </div>
        </div>
    </div>
</div>
```

```
<?php include 'footer.php'; ?>
```

Oke coba kita lihat hasilnya. Klik pada tombol tambah. Maka form tambah pelanggan yang terletak pada file pelanggan\_tambah.php di atas akan di tampilkan.



Form untuk input pelanggan pun selesai. Tapi form ini belum selesai sepenuhnya. Kita belum membuat file php yang menangani data yang di input pada form ini.

Perhatikan pada tag <form> nya.

```
<form method="post" action="pelanggan_aksi.php">
```

Perhatikan di sini. Di sini kita akan memproses data yang diinput pada form tambah pelanggan di atas pada file pelanggan\_aksi.php.

Jadi sekarang kita buat sebuah file baru dalam folder admin dengan nama "pelanggan\_aksi.php". pada file pelanggan\_aksi.php ini kita akan membuat proses atau aksi untuk menyimpan data yang di input ke dalam database (tabel pelanggan).

Pelanggan\_aksi.php

```
<?php
```

```
// menghubungkan dengan koneksi
include '../koneksi.php';

// menangkap data yang dikirim dari form
$nama = $_POST['nama'];
$hp = $_POST['hp'];
$alamat = $_POST['alamat'];

// input data ke tabel pelanggan
mysqli_query($koneksi,"insert into pelanggan
values('','$nama','$hp','$alamat')");

header("location:pelanggan.php");

?>
```

Perhatikan pada syntax pelanggan\_aksi.php di atas.

Kita menggunakan query sql sebagai berikut untuk menyimpan data yang di input masuk ke database.

```
mysqli_query($koneksi,"insert into pelanggan
values('','$nama','$hp','$alamat')");
```

Penjelasan tentang query seperti di atas telah di jelaskan pada sub judul sebelumnya. Silahkan teman-teman baca lagi jika masih bingung.

Sekarang kita akan mencoba menginput data pelanggan baru dengan nama samsul, nomor hp dan alamat nya sesuai seperti gambar di bawah.

localhost/laundry/admin/pelanggan\_tambah.php

LAUNDRY Malas Ngoding

Halo, admin !

Tambah Pelanggan Baru

Nama  
Samsul

HP  
0887363534555

Alamat  
Jl. Merpati no.34

Simpan

Lalu klik tombol simpan.

localhost/laundry/admin/pelanggan.php

LAUNDRY Malas Ngoding

Halo, admin !

Data Pelanggan

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Samsul	0887363534555	Jl. Merpati no.34	<a href="#">Edit</a> <a href="#">Hapus</a>

Data pelanggan baru yang di input pun berhasil tersimpan dan tampil pada tabel data pelanggan seperti gambar di atas.

## Mengedit Data Pelanggan

Setelah kita selesai membuat proses menampilkan data pelanggan dari tabel pelanggan dan meginput data pelanggan baru ke tabel pelanggan. Pada sub judul ini kita akan belajar cara mengedit data pelanggan.

Perhatikan link edit yang sudah kita buat pada tabel yang menampilkan data pelanggan pada file pelanggan.php. terdapat link yang kita buat yang mengarahkan ke file pelanggan\_edit.php sambil mengirimkan data id si pelanggan yang ingin di edit.

```
<a href="pelanggan_edit.php?id=<?php echo $d['pelanggan_id']; ?>" class="btn btn-sm btn-info">Edit</a>
```

Karena pada file pelangan\_edit.php ini kita akan membuat form untuk mengedit data pelanggan.

Buat sebuah file baru dengan nama pelanggan\_edit.php.

### Pelanggan\_edit.php

```
<?php include 'header.php'; ?>

<div class="container">
    <br/>
    <br/>
    <br/>
    <div class="col-md-5 col-md-offset-3">

        <div class="panel">
            <div class="panel-heading">
                <h4>Edit Pelanggan</h4>
            </div>
            <div class="panel-body">

                <?php
                // menghubungkan koneksi
                include '../koneksi.php';

                // menangkap id yang dikirim melalui url
                $id = $_GET['id'];

                // megambil data pelanggan yang ber id di atas dari
                tabel pelanggan
                $data = mysqli_query($koneksi,"select * from
                pelanggan where pelanggan_id='$id'");
                while($d=mysqli_fetch_array($data)) {
                    ?>

                <form method="post"
```

```

action="pelanggan_update.php">
    <div class="form-group">
        <label>Nama</label>
        <!-- form id pelanggan yang di
edit, untuk di kirim ke file aksi -->
        <input type="hidden" name="id"
value="<?php echo $d['pelanggan_id']; ?>">
        <input type="text" class="form-
control" name="nama" placeholder="Masukkan nama .." value="<?php echo
$d['pelanggan_nama']; ?>">
    </div>

    <div class="form-group">
        <label>HP</label>
        <input type="number" class="form-
control" name="hp" placeholder="Masukkan no.hp .." value="<?php echo
$d['pelanggan_hp']; ?>">
    </div>

    <div class="form-group">
        <label>Alamat</label>
        <input type="text" class="form-
control" name="alamat" placeholder="Masukkan alamat .." value="<?php echo
$d['pelanggan_alamat']; ?>">
    </div>

    <br/>

    <input type="submit" class="btn btn-
primary" value="Simpan">
</form>

<?php
}
?>
</div>
</div>
</div>

<?php include 'footer.php'; ?>

```

Form edit pelanggan ini kita buat dengan menetapkan file “pelanggan\_update.php” sebagai file yang mengelola data yang dikirim dari form edit pelanggan ini.

```
<form method="post" action="pelanggan_update.php">
```

Jadi sekarang kita buat file baru lagi dengan nama “pelanggan\_update.php”. buat di dalam folder admin.

## **pelanggan\_update.php**

```
<?php
// menghubungkan koneksi
include '../koneksi.php';

// menangkap data yang dikirim dari form
$id = $_POST['id'];
$nama = $_POST['nama'];
$hp = $_POST['hp'];
$alamat = $_POST['alamat'];

// update data
mysqli_query($koneksi,"update pelanggan set pelanggan_nama='$nama',
pelanggan_hp='$hp', pelanggan_alamat='$alamat' where pelanggan_id='$id'");

// mengalihkan halaman kembali ke halaman pelanggan
header("location:pelanggan.php");

?>
```

Seperti yang teman-teman bisa lihat pada pelanggan\_update.php di atas. Pertama kita hubungkan dengan koneksi database. Kemudian kita tangkap data yang dikirim dari form edit pelanggan. Kemudian tingal kita update data pelanggan dengan query

```
mysqli_query($koneksi,"update pelanggan set pelanggan_nama='$nama',
pelanggan_hp='$hp', pelanggan_alamat='$alamat' where pelanggan_id='$id'");
```

Cara nya sama saja dengan tutorial crud dengan PHP dan MySQLi yang pernah dijelaskan pada sub judul sebelumnya. Sekarang kita coba mengedit data pelanggan dengan klik pada tombol edit.

localhost/laundry/admin/pelanggan.php

LAUNDRY Malas Ngoding

Halo, admin !

Data Pelanggan

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Samsul	0887363534555	Jl. Merpati no.34	<a href="#">Edit</a> <a href="#">Hapus</a>

Disini saya mencoba untuk mengedit data pelanggan nomor 3, yaitu samsul. Setelah klik tombol edit maka akan muncul form edit pelanggan.

localhost/laundry/admin/pelanggan\_edit.php?id=3

LAUNDRY Malas Ngoding

Halo, admin !

Edit Pelanggan

**Nama**  
Samsul

**HP**  
887363534555

**Alamat**  
Jl. Merpati no.34

[Simpan](#)

Data si samsul pun tampil dalam form. Tinggal langsung kita coba ubah. Saya mencoba mengubah nama "samsul" dengan "jamaludin".

Edit Pelanggan

**Nama**  
Jamaludin

**HP**  
887363534555

**Alamat**  
Jl. Merpati no.34

**Simpan**

Dan klik pada tombol simpan.

Data Pelanggan

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Jamaludin	0887363534555	Jl. Merpati no.34	<a href="#">Edit</a> <a href="#">Hapus</a>

Maka data pelanggan pun berhasil berubah. Cukup mudah bukan. Selanjutnya akan kita lanjutkan dengan cara menghapus data pelanggan dengan php dan mysqli.

## Menghapus Data Pelanggan

Pada sub judul ini kita akan belajar cara membuat tombol yang bisa menghapus data pelanggan. Perhatikan pada pelanggan.php. di sana kita telah membuat tombol hapus.

```
<a href="pelanggan_hapus.php?id=<?php echo $d['pelanggan_id']; ?>" class="btn btn-sm btn-danger">Hapus</a>
```

Yang mengarah ke pelanggan\_hapus.php, sambil mengirimkan data id si pelanggan yang ingin di hapus. Buat sebuah file php baru lagi dalam folder admin, dengan nama "pelanggan\_hapus.php".

### Pelanggan\_hapus.php

```
<?php
// menghubungkan koneksi
include '../koneksi.php';

// menangkap data id yang dikirim dari url
$id = $_GET['id'];

// menghapus pelanggan
mysqli_query($koneksi,"delete from pelanggan where pelanggan_id='$id'");

// alihkan halaman ke halaman pelanggan
header("location:pelanggan.php");
?>
```

Sekarang coba klik pada tombol hapus. Di sini saya coba klik hapus pada data pelanggan yang nomor 3, yaitu jamaludin.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/pelanggan.php`. The page title is "Data Pelanggan". A table lists three customers: Sumanto, Sri Dayatun, and Jamaludin. Each row has "Edit" and "Hapus" buttons. A red arrow points to the "Hapus" button for Jamaludin.

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Jamaludin	0887363534555	Jl. Merpati no.34	<a href="#">Edit</a> <a href="#">Hapus</a>

Maka pelanggan yang bernama jamaludin pun di hapus.

The screenshot shows the same web application after the deletion. The table now only displays two rows: Sumanto and Sri Dayatun. A red arrow points to the empty space where the third row (Jamaludin) used to be.

No	Nama	HP	Alamat	OPSI
1	Sumanto	08123536738	Jl. cenderawasih no.126 bandung	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Sri Dayatun	08736363444	Jl. tangkuban perahu no.23 Bandung	<a href="#">Edit</a> <a href="#">Hapus</a>

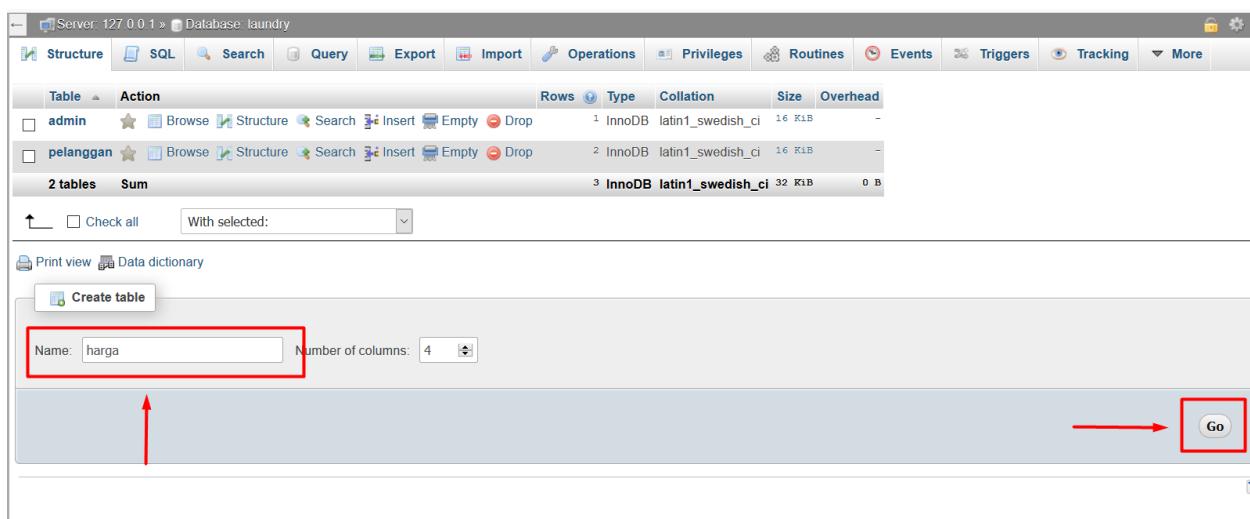
Oke selesai. Data pelanggan berhasil di hapus.

# Membuat Pengaturan Harga Laundry Per Kilo

Pengaturan harga per kilo ini merupakan sebuah fitur unggulan dari sistem informasi laundry yang kita buat. Jadi admin bisa menetapkan berapa harga per kilo dari cucian yang dibawa oleh pelanggan. Misalnya perkilo admin menetapkan harga 10.000 rupiah, atau 6.000 rupiah. Nanti admin bisa menentukan sendiri.

## Membuat Table Harga

Untuk cara membuatnya, kita memerlukan sebuah tabel dalam database laundry yang bisa menyimpan data harga per kilo. Untuk itu kita akan buat sebuah tabel baru dengan nama “harga”.



Selanjutnya akan muncul form berikut dan isi sesuai gambar berikut.

Server: 127.0.0.1 > Database: laundry > Table: harga

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table name: harga Add 1 column(s) Go

**Structure**

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.	Comments	Virtu
harga_per_kilo	INT		None							
Pick from Central Columns										
Pick from Central Columns										
Pick from Central Columns										
Pick from Central Columns										
Table comments:				Collation:	Storage Engine: InnoDB					
PARTITION definition:										
Console										

Preview SQL Save

Buat column nya dengan nama “harga\_per\_kilo”. Biarkan ber tipe int. dan langsung klik save.

Server: 127.0.0.1 > Database: laundry > Table: harga

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table structure Relation view

**Insert**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	harga_per_kilo	int(11)		No None				Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns

Check all With selected: Browse Change Drop Primary Unique Index Add to central columns Remove from central columns

Print view Propose table structure Track table Move columns Improve table structure

Add 1 column(s) after harga\_per\_kilo Go

+ Indexes

Information

Space usage	Row statistics
Data 16 KiB	Format Compact
Index 0 B	Collation latin1_swedish_ci
Total 16 KiB	Creation Apr 05, 2018 at 02:34 PM

Setelah tabel harga selesai di buat, selanjutnya kita akan menginput harga baru, dengan cara klik pada menu insert.

Server: 127.0.0.1 » Database: laundry » Table: harga

**Column Type Function Null Value**

harga_per_kilo	int(11)		10000
----------------	---------	--	-------

Ignore

**Column Type Function Null Value**

harga_per_kilo	int(11)		
----------------	---------	--	--

**Go**

Insert as new row and then Go back to previous page

Dan masukkan harga per kilo nya sesuai dengan keinginan teman-teman. Dan klik go.

Server: 127.0.0.1 » Database: laundry » Table: harga

**Browse Structure SQL Search Insert Export Import Privileges**

**Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.**

Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)

```
SELECT * FROM `harga`
```

Show all Number of rows: 25 Filter rows: Search this table

+ Options
harga_per_kilo
10000

Nah harga 10.000 pun berhasil tersimpan dalam tabel harga. ingat, cukup 1 record saja untuk harga per kilonya. Jangan di tambahkan lagi. Karena pada tabel harga ini kita menyimpan harga penjualan laundry per kilo.

## Membuat Pengaturan Harga Laundry

Perhatikan pada menu “pengaturan harga”. saat di klik pada menu tersebut, halaman akan di alihkan ke harga.php.

Sekarang buat file baru dalam folder admin dengan nama harga.php. pada file harga.php ini kita akan membuat form untuk pengaturan harga laundry.

### Harga.php

```
<?php include 'header.php'; ?>

<div class="container">
    <br/>
    <br/>
    <br/>
    <div class="col-md-5 col-md-offset-3">

        <div class="panel">
            <div class="panel-heading">
                <h4>Pengaturan Harga Laundry</h4>
            </div>
            <div class="panel-body">

                <?php
                // menghubungkan koneksi
                include '../koneksi.php';

                // megambil data haraga per kilo dari tabel harga
                $data = mysqli_query($koneksi,"select harga_per_kilo
from harga");
                while($d=mysqli_fetch_array($data)) {
                    ?>

                    <form method="post" action="harga_update.php">
                        <div class="form-group">
                            <label>Harga per kilo</label>
                            <input type="number" class="form-control" name="harga" value="<?php echo $d['harga_per_kilo']; ?>">
                        </div>

                        <br/>

                        <input type="submit" class="btn btn-primary" value="Ubah Harga">
                    </form>

                <?php
                }
                ?>
            </div>
        </div>
    </div>
```

```
</div>  
</div>  
<?php include 'footer.php'; ?>
```

Pada file harga.php di atas, kita menampilkan harga yang sudah kita input sebelumnya ke tabel harga. harga yang di tampilkan dalam form, bisa di ubah dan di update.

Pada form pengaturan harga ini, aksinya telah kita tetapkan pada file harga\_update.php

### Harga\_update.php

```
<form method="post" action="harga_update.php">
```

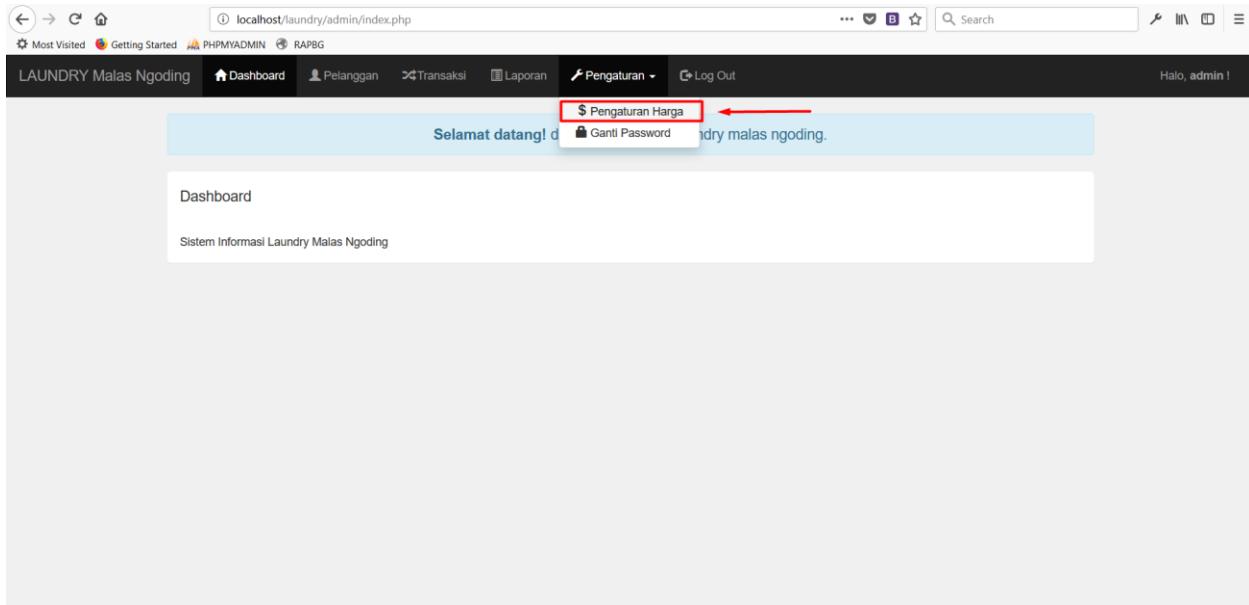
Jadi kita buat file baru lagi dengan nama harga\_update.php.

### Harga\_update.php

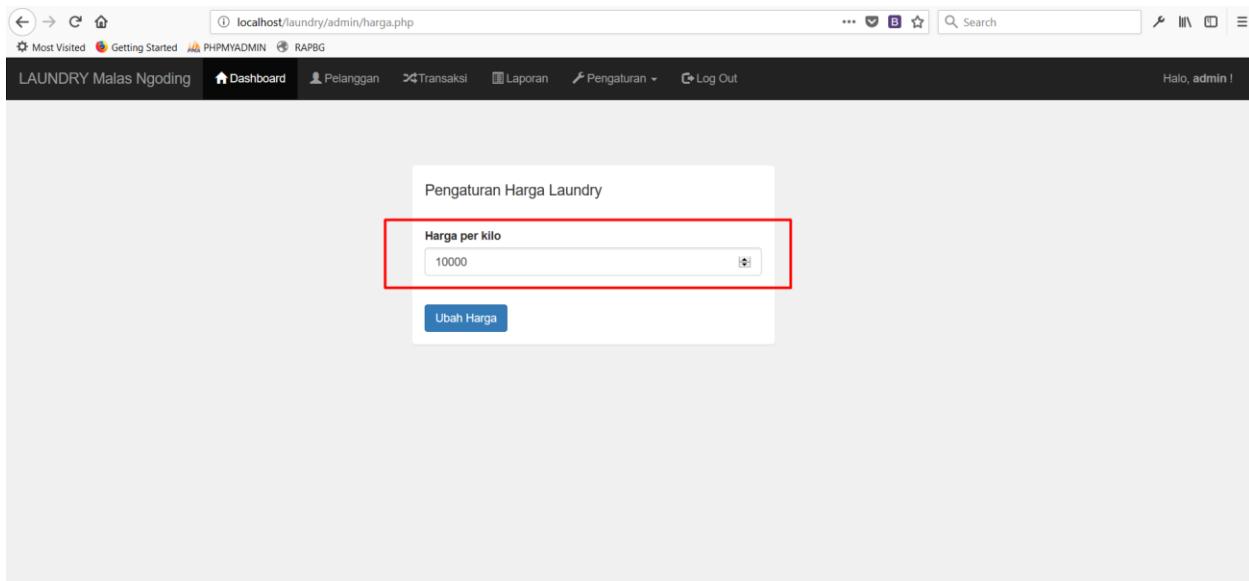
```
<?php  
// menghubungkan koneksi  
include '../koneksi.php';  
  
// menangkap data yang dikirim dari form  
$harga = $_POST['harga'];  
  
// update data  
mysqli_query($koneksi,"update harga set harga_per_kilo='$harga'");  
  
// mengalihkan halaman kembali ke halaman pelanggan  
header("location:harga.php");  
?>
```

Penjelasannya sama saja seperti membuat update data pada tutorial CRUD sebelumnya.

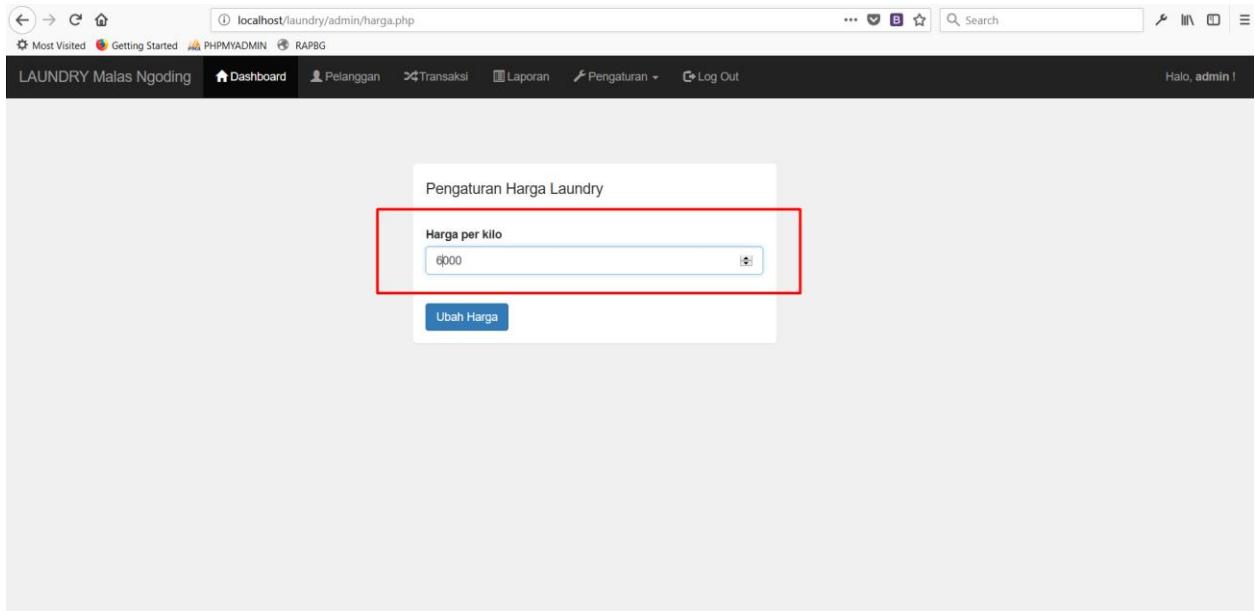
Sekarang kita coba klik pada menu pengaturan harga.



Maka kan muncul form pengaturan harga.



Sekarnang coba kita ubah harga nya dari 10.000 menjadi 6.000. dan klik ubah harga.



Nah, maka harga per kilo pun berhasil berubah.

Sampai di sini kita telah selesai membuat pengaturan harga. tinggal nanti ke depannya akan kita integrasikan harga per kilo ini dengan jumlah berat cucian yang dibawa oleh pelanggan saat transaksi.

## Membuat Transaksi Laundry

Akhirnya sampai juga kita pada tahap membuat transaksi laundry pada sistem informasi laundry ini. Transaksi yang akan kita buat ini adalah transaksi yang dilakukan pada saat pelanggan datang dan membawa cuciannya untuk di cuci.

Kemudian admin akan menginput data pelanggan, tanggal cucian selesai, mengisi berat cucian dan harganya akan muncul dengan menghitung berat cucian dikalikan dengan harga cucian per kilo nya sesuai dengan pada pengaturan harga laundry yang telah kita buat sebelumnya.

Kita juga bisa membuat detail laundry, berupa jenis pakaian dan jumlah pakaian tersebut. Kemudian admin juga bisa mencetakkan invoice untuk si pelanggan. Sebagai bukti laundry yang harus dibawa pada saat ingin mengambil cuciannya kembali.

Langsung saja kita mulai ya. Kita butuh 2 buah tabel baru dalam database laundry untuk menyimpan data transaksi dan mengimpaun data cucian si pelanggan. Oleh sebab itu kita akan membuat tabel baru dengan nama “transaksi” dan “pakaian”. Yuk simak caranya.

### Membuat Table Transaksi

Sesuai dengan perancangan database sistem informasi laundry yang telah dibahas pada sub judul sebelumnya. Pada table transaksi ini kita akan menyimpan data-data transaksi, termasuk nomor invoice si pelanggan juga akan kita ambil dari id transaksi pada table ini.

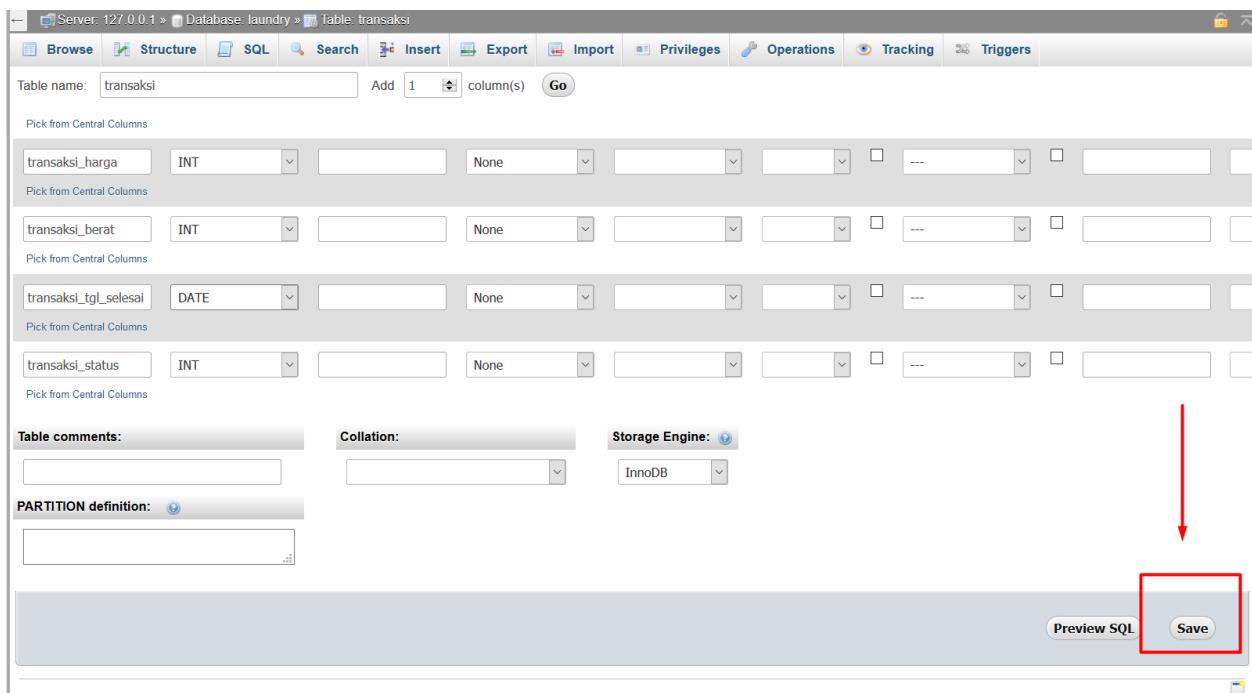
Buat sebuah table baru dengan nama “transaksi”.

Isi nama tabel yang akan di buat dengan “transaksi”. Dan jumlah kolumn yang kita butuhkan “7”. Klik go.

Pada kolumnnya isikan nama kolumnnya dengan transaksi\_id dan bertipe int, jangan lupa klik auto\_increment pada transaksi\_id. Kemudian transaksi\_tgl dengan tipe data date, transaksi\_pelanggan dengan tipe data int, transaksi\_harga dengan tipe data int, transaksi\_berat dengan tipe data tgl\_selesai dengan tipe data date dan terakhir transaksi\_status dengan tipe data int.

Pada transaksi\_tgl kita akan menyimpan tanggal laundry pada transaksi\_pelanggan kita akan menyimpan id pelanggan yang laundry, pada transaksi\_harga kita akan menyimpan harga cucian si pelanggan, pada transaksi\_berat kita menyimpan berat cucian di pelanggan, transaksi\_tgl\_selesai kita menyimpan tanggal cucian tersebut selesai dan bisa di ambil oleh pelanggan, dan pada transaksi\_status kita akan menyimpan status cucian atau status laundry tersebut. Kita akan menyimpan data berupa angka pada kolumn transaksi\_status. Misalnya 0 = di proses, 1 = di cuci, dan 2 = selesai. Jika berstatus 0 berarti cucian masih dalam proses, jika berstatus 1 berarti cucian telah selesai di cuci dan siap untuk di ambil, dan jika berstatus 2 berarti cucian telah selesai di ambil oleh pelanggan dan transaksi di anggap telah selesai. Oke lanjut.

Jika telah selesai, klik save.



Maka table selesai dibuat.

The screenshot shows the 'Table structure' view for the 'transaksi' table. The table has 7 columns:

- transaksi\_id**: int(11), No, None, AUTO\_INCREMENT
- transaksi\_tgl**: date
- transaksi\_pelanggan**: int(11)
- transaksi\_harga**: int(11)
- transaksi\_berat**: int(11)
- transaksi\_tgl\_selesai**: date
- transaksi\_status**: int(11)

Action buttons for each column include Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values, and More.

## Membuat Table Pakaian

Setelah selesai membuat table transaksi, kita lanjutkan dengan membuat table pakaian untuk menyimpan data pakaian dalam transaksi yang di lakukan saat pelanggan menyerahkan cuciannya.

Buat table baru dengan nama “pakaian”.

The screenshot shows the 'Structure' tab of the MySQL Workbench interface. It lists four tables in the database:

- admin
- harga
- pelanggan
- transaksi

A 'Create table' dialog is open, showing the table name 'pakaian' and the number of columns set to 4.

Isi sesuai dengan gambar berikut.

Table name: pakaian

**Structure**

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.	Comments	Virtua
pakaian_id	INT		None					<input checked="" type="checkbox"/> PRIMARY		
pakaian_transaksi	INT		None							
pakaian_jenis	VARCHAR	255	None							
pakaian_jumlah	INT		None							

Table comments:

Collation: Storage Engine: InnoDB

PARTITION definition:

Preview SQL Save

Pada kolumn pakaian\_id kita menyimpan id pakaian, pada pakaian\_transaksi ini kita akan menyimpan id transaksi, pada pakaian\_jenis kita menyimpan jenis pakaian misalnya seperti celana jeans, baju kaos, baju kemeja dan lain-lain. Dan terakhir pada pakaian\_jumlah kita menyimpan data jumlah pakaian, misalnya celana jeans jumlahnya 2, baju kaos jumlahnya 1 dan lain-lain. Jangan lupa untuk mencentang pada auto\_increment pada pakaian\_id. Dan klik save.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	pakaian_id	int(11)		No	None	AUTO_INCREMENT		<input type="checkbox"/> Change <input type="checkbox"/> Drop <input checked="" type="checkbox"/> Primary <input type="checkbox"/> Unique <input type="checkbox"/> Index <input type="checkbox"/> Spatial <input type="checkbox"/> Fulltext <input type="checkbox"/> Distinct value
2	pakaian_transaksi	int(11)		No	None			<input type="checkbox"/> Change <input type="checkbox"/> Drop <input checked="" type="checkbox"/> Primary <input type="checkbox"/> Unique <input type="checkbox"/> Index <input type="checkbox"/> Spatial <input type="checkbox"/> Fulltext <input type="checkbox"/> Distinct value
3	pakaian_jenis	varchar(255)		No	None			<input type="checkbox"/> Change <input type="checkbox"/> Drop <input checked="" type="checkbox"/> Primary <input type="checkbox"/> Unique <input type="checkbox"/> Index <input type="checkbox"/> Spatial <input type="checkbox"/> Fulltext <input type="checkbox"/> Distinct value
4	pakaian_jumlah	int(11)		No	None			<input type="checkbox"/> Change <input type="checkbox"/> Drop <input checked="" type="checkbox"/> Primary <input type="checkbox"/> Unique <input type="checkbox"/> Index <input type="checkbox"/> Spatial <input type="checkbox"/> Fulltext <input type="checkbox"/> Distinct value

Check all With selected: Browse Change Drop Primary Unique Index Add to central columns Remove from central columns

Print view Propose table structure Track table Move columns Improve table structure

Add 1 column(s) after pakaian\_jumlah Go

+ Indexes

Information

Dan tabel pakaian pun selesai kita buat.

## Menampilkan Data Transaksi Laundry

Perhatikan pada menu transaksi, pada menu transaksi kita mengarahkan ke file transaksi.php pada saat di klik.

```
<li><a href="transaksi.php"><i class="glyphicon glyphicon-random"></i>  
Transaksi</a></li>
```

Pada file transaksi.php kita akan menampilkan data transaksi. Silahkan ketik atau copas syntax berikut.

### Transaksi.php

```
<?php include 'header.php'; ?>

<div class="container">
    <div class="panel">
        <div class="panel-heading">
            <h4>Data Transaksi Laundry</h4>
        </div>
        <div class="panel-body">

            <a href="transaksi_tambah.php" class="btn btn-sm btn-info pull-right">Transaksi Baru</a>

            <br/>
            <br/>

            <table class="table table-bordered table-striped">
                <tr>
                    <th width="1%">No</th>
                    <th>Invoice</th>
                    <th>Tanggal</th>
                    <th>Pelanggan</th>
                    <th>Berat (Kg)</th>
                    <th>Tgl. Selesai</th>
                    <th>Harga</th>
                    <th>Status</th>
                    <th width="20%">OPSI</th>
                </tr>
                <?php
                // koneksi database
                include '../koneksi.php';

                // mengambil data pelanggan dari database
                $data = mysqli_query($koneksi,"select * from
pelanggan,transaksi      where      transaksi_pelanggan=pelanggan_id      order      by
transaksi_id desc");
                $no = 1;
                // mengubah data ke array dan menampilkannya dengan
perulangan while
                while($d=mysqli_fetch_array($data)) {
                    ?>
```

```

        <tr>
            <td><?php echo $no++; ?></td>
            <td>INVOICE-<?php echo
$ď['transaksi_id']; ?></td>
            <td><?php echo $d['transaksi_tgl'];
?></td>
            <td><?php echo $d['pelanggan_nama'];
?></td>
            <td><?php echo $d['transaksi_berat'];
?></td>
            <td><?php echo
$d['transaksi_tgl_selesai']; ?></td>
            <td><?php echo "Rp.
".number_format($d['transaksi_harga'])." , -"; ?></td>
            <td>
                <?php
                if($d['transaksi_status']=="0"){
                    echo "<div class='label
label-warning'>PROSES</div>";
                } else
                if($d['transaksi_status']=="1"){
                    echo "<div class='label
label-info'>DICUCI</div>";
                } else
                if($d['transaksi_status']=="2"){
                    echo "<div class='label
label-success'>SELESAI</div>";
                }
            ?>

            </td>
            <td>
                <a href="transaksi_invoice.php?id=<?php echo $d['transaksi_id']; ?>" target="_blank" class="btn btn-sm btn-warning">Invoice</a>
                <a href="transaksi_edit.php?id=<?php echo $d['transaksi_id']; ?>" class="btn btn-sm btn-info">Edit</a>
                <a href="transaksi_hapus.php?id=<?php echo $d['transaksi_id']; ?>" class="btn btn-sm btn-danger">Batalkan</a>
            </td>
        </tr>
    <?php
}
?>
</table>
</div>
</div>
<?php include 'footer.php'; ?>

```

Secara keseluruhan, cara menampilkan data transaksi ini masih sama seperti pada tutorial CRUD yang telah dijelaskan sebelumnya. Tapi akan saya jelaskan sedikit perbedaannya.

Di sini kita menampilkan data dari 2 table sekaligus. Perhatikan query di atas.

```
$data = mysqli_query($koneksi,"select * from pelanggan,transaksi where  
transaksi_pelanggan=pelanggan_id order by transaksi_id desc");
```

Di sini kita menggunakan query untuk menampilkan semua data dari table pelanggan dan transaksi yang data transaksi\_pelanggan nya sama dengan id pelanggan, dan kita urutkan dari id transaksi terbesar ke terkecil.

Seperti yang telah dijelaskan sebelumnya, pada kolumn transaksi\_pelanggan dalam table transaksi kita menyimpan id dari pelanggan.

Dan seperti yang telah dijelaskan juga sebelumnya tentang status transaksi,

```
<?php  
if($d['transaksi_status']=="0") {  
    echo "<div class='label label-warning'>PROSES</div>";  
} else if($d['transaksi_status']=="1") {  
    echo "<div class='label label-info'>DICUCI</div>";  
} else if($d['transaksi_status']=="2") {  
    echo "<div class='label label-success'>SELESAI</div>";  
}  
?>
```

Pada kolumn status transaksi kita membuat pengecekan, jika status transaksi nya 0, maka kita tampilkan “PROSES”. Jika status transaksinya sama dengan 1, maka kita tampilkan “DICUCI”. Jika status transaksi nya sama dengan 3, maka kita tampilkan “SELESAI”.

Kita lihat hasilnya.

Klik pada menu transaksi.

Selesai, tapi di sini datanya masih kosong, karena kita belum menginput data ke table transaksi. Untuk itu akan kita lanjutkan ke cara membuat transaksi baru pada sistem informasi laundry.

## Menginput Data Transaksi Laundry Baru

Pada tombol “transaksi baru” di atas sudah kita letakkan link ke “transaksi\_tambah.php” saat di klik.

```
<a href="transaksi_tambah.php" class="btn btn-sm btn-info pull-right">Transaksi Baru</a>
```

Pada transaksi\_tambah.php inilah akan kita buat form penginputan data transaksi baru. Langsung saja buat sebuah file baru dalam folder admin dengan nama “transaksi\_tambah.php”.

```
<?php include 'header.php'; ?>

<?php

// koneksi database
include '../koneksi.php';
?>
<div class="container">
    <div class="panel">
        <div class="panel-heading">
            <h4>Transaksi Laundry Baru</h4>
        </div>
```

```

<div class="panel-body">

    <div class="col-md-8 col-md-offset-2">
        <a href="transaksi.php" class="btn btn-sm btn-info pull-right">Kembali</a>
        <br/>
        <br/>
        <form method="post" action="transaksi_aksi.php">
            <div class="form-group">
                <label>Pelanggan</label>
                <select class="form-control" name="pelanggan" required="required">
                    <option value="">- Pilih Pelanggan</option>
                    <?php
                        // mengambil data pelanggan dari database
                        $data =
                        mysqli_query($koneksi,"select * from pelanggan");
                    // mengubah data ke array dan menampilkannya dengan perulangan while
                    while($d=mysqli_fetch_array($data)){
                        ?>
                        <option value="<?php echo $d['pelanggan_id']; ?>"><?php echo $d['pelanggan_nama']; ?></option>
                    }
                    ?>
                </select>
            </div>

            <div class="form-group">
                <label>Berat</label>
                <input type="number" class="form-control" name="berat" placeholder="Masukkan berat cucian .." required="required">
            </div>

            <div class="form-group">
                <label>Tgl. Selesai</label>
                <input type="date" class="form-control" name="tgl_selesai" required="required">
            </div>

            <br/>

            <table class="table table-bordered table-striped">
                <tr>
                    <th>Jenis Pakaian</th>
                    <th width="20%">Jumlah</th>
                </tr>

```



```

                <td><input type="number"
class="form-control" name="jumlah_pakaian[]"></td>
            </tr>
        </table>

        <input type="submit" class="btn btn-primary"
value="Simpan">
    </form>

</div>

</div>
</div>
<?php include 'footer.php'; ?>
```

Pada transaksi\_tambah.php di atas kita membuat sebuah form untuk penginputan data transaksi laundry baru seperti penginputan pelanggan, berat, pakaian yang di laundry dan lain-lain.

Perhatikan pada form tersebut. Pada form pelanggan kita tampilkan semua data pelanggan dalam form dropdown. Sehingga bisa di pilih langsung oleh admin.

```

<div class="form-group">
    <label>Pelanggan</label>
    <select class="form-control" name="pelanggan"
required="required">
        <option value="">- Pilih Pelanggan</option>

        <?php
        // mengambil data pelanggan dari database
        $data = mysqli_query($koneksi,"select * from pelanggan");

        // mengubah data ke array dan menampilkannya dengan
perulangan while
        while($d=mysqli_fetch_array($data)) {
        ?>
            <option value="<?php echo $d['pelanggan id']; ?>"><?php
echo $d['pelanggan_nama']; ?></option>

            <?php
        }
        ?>
    </select>
</div>
```

Kemudian pada form selanjutnya kita buat form berat dan tgl selesai. Untuk menyimpan berapa jumlah berat cucian si pelanggan dan tanggal berapa cucian tersebut selesai.

```
<div class="form-group">
    <label>Berat</label>
    <input type="number" class="form-control" name="berat"
placeholder="Masukkan berat cucian .." required="required">
</div>

<div class="form-group">
    <label>Tgl. Selesai</label>
    <input type="date" class="form-control" name="tgl_selesai"
required="required">
</div>
```

Dan terakhir kita juga membuat form untuk penginputan jenis pakaian dan jumlah masing-masingnya sebanyak 10 form.

Seperti yang teman-teman lihat, pada form penginputan jenis dan jumlah pakaian, kita menggunakan tanda “ [ ] ” pada nama form. Agar form tersebut menjadi form array. Sehingga memudahkan kita menangkap data form yang berjenis banyak seperti ini nantinya pada file aksi.

```
<td><input type="text" class="form-control" name="jenis_pakaian[]"></td>
<td><input type="number" class="form-control" name="jumlah_pakaian[]"></td>
```

Cara menangkap data form array seperti ini akan di jelaskan pada file aksi dari form input transaksi baru ini.

Perhatikan tag `<form>` pada `transaksi_tambah.php` di atas, kita menetapkan file aksi nya di “`transaksi_aksi.php`”.

```
<form method="post" action="transaksi_aksi.php">
```

Buat sebuah file baru dalam folder admin, dengan nama `transaksi_aksi.php`.

`Transaksi_aksi.php`

```
<?php
// menghubungkan dengan koneksi
include '../koneksi.php';

// menangkap data yang dikirim dari form
```

```

$pelanggan = $_POST['pelanggan'];
$berat = $_POST['berat'];
$tgl_selesai = $_POST['tgl_selesai'];

$tgl_hari_ini = date('Y-m-d');
$status = 0;

// mengambil data harga per kilo dari database
$h = mysqli_query($koneksi,"select harga_per_kilo from harga");
$harga_per_kilo = mysqli_fetch_assoc($h);

// menghitung harga laundry, harga perkilo x berat cucian
$harga = $berat * $harga_per_kilo['harga_per_kilo'];

// input data ke tabel transaksi
mysqli_query($koneksi,"insert into transaksi
values('','$tgl_hari_ini','$pelanggan','$harga','$berat','$tgl_selesai','$status')");

// menyimpan id dari data yang di simpan pada query insert data sebelumnya
$id_terakhir = mysqli_insert_id($koneksi);

// menangkap data form input array (jenis pakaian dan jumlah pakaian)
$jenis_pakaian = $_POST['jenis_pakaian'];
$jumlah_pakaian = $_POST['jumlah_pakaian'];

// input data cucian berdasarkan id transaksi (invoice) ke table pakaian
for($x=0;$x<count($jenis_pakaian);$x++) {
    if($jenis_pakaian[$x] != "") {
        mysqli_query($koneksi,"insert into pakaian
values('','$id_terakhir','$jenis_pakaian[$x]','$jumlah_pakaian[$x]')");
    }
}

header("location:transaksi.php");
?>

```

Kita jalankan dulu untuk melihat hasilnya. Baru kemudian akan saya jelaskan tahap-tahap yang telah kita buat di atas.

Klik pada menu transaksi. Dan klik pada “transaksi baru”.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/transaksi.php`. The page title is "LAUNDRY Malas Ngoding". The navigation menu includes "Dashboard", "Pelanggan", "Transaksi", "Laporan", "Pengaturan", and "Log Out". A user "Halo, admin!" is logged in. The main content area is titled "Data Transaksi Laundry" and contains a table header with columns: No, Invoice, Tanggal, Pelanggan, Berat (Kg), Tgl. Selesai, Harga, Status, and Opsi. A blue button labeled "Transaksi Baru" is visible at the top right of the table area.

Maka akan muncul form penginputan transaksi baru.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/transaksi_tambah.php`. The page title is "LAUNDRY Malas Ngoding". The navigation menu includes "Dashboard", "Pelanggan", "Transaksi", "Laporan", "Pengaturan", and "Log Out". A user "Halo, admin!" is logged in. The main content area is titled "Transaksi Laundry Baru" and contains several input fields: "Pelanggan" (dropdown menu with placeholder "- Pilih Pelanggan"), "Berat" (text input field with placeholder "Masukkan berat cucian .."), "Tgl. Selesai" (text input field with placeholder "mm / dd / yyyy"). Below these is a table with two columns: "Jenis Pakai" and "Jumlah". There are four rows in the table, each with a dropdown arrow icon in the "Jumlah" column.

Tgl. Selesai  
mm / dd / yyyy

Jenis Pakaian	Jumlah

Simpan

Coba kita input data transaksi laundry baru. Disini pelanggan yang saya pilih misalnya sumanto,

Pelanggan  
- Pilih Pelanggan  
- Pilih Pelanggan  
Sumanto  
Sri Dayatun

Kembali

Tgl. Selesai  
mm / dd / yyyy

Jenis Pakaian	Jumlah

Dan beratnya 2 kg, karena sekarang tanggal 6 april, katakanlah misalnya lama laundry nya 3 hari, sehingga tanggal selesainya saya input tanggal 9 april. Seperti pada gambar berikut.

The screenshot shows a web-based laundry management system. At the top, there's a navigation bar with links like 'Dashboard', 'Pelanggan', 'Transaksi', 'Laporan', 'Pengaturan', and 'Log Out'. A user 'Halo, admin!' is logged in. The main area is titled 'Transaksi Laundry Baru'. It contains several input fields: 'Pelanggan' (containing 'Sumantri'), 'Berat' (containing '2'), 'Tgl. Selesai' (containing '04 / 09 / 2018'), and a table for 'Jenis Pakaian' (Clothing Types) with rows for 'Celana Jeans' (2), 'Baju Kemeja' (1), and 'Baju Kaos' (4). Each row has an 'input' field for the item name and a 'dropdown' field for the quantity. Red arrows point from the text 'Kemudian juga mengisi pada jenis-jenis pakaian apa saja. Yaitu ada celana jeans sebanyak 2 buah, baju kemeja 1, dan baju kaos sebanyak 4 buah.' to each of these three rows.

Kemudian juga mengisi pada jenis-jenis pakaian apa saja. Yaitu ada celana jeans sebanyak 2 buah, baju kemeja 1, dan baju kaos sebanyak 4 buah.

Kemudian klik simpan.

This screenshot shows the same laundry transaction form after the data has been saved. The 'Tgl. Selesai' field still contains '04 / 09 / 2018'. The 'Jenis Pakaian' table now has 12 empty rows, indicating that the previous entries have been processed. A red arrow points from the text 'Maka data transaksi selesai dan berhasil di input ke database.' to the 'Simpan' button at the bottom of the form.

Maka data transaksi selesai dan berhasil di input ke database.

Seperti yang teman-teman lihat pada gambar di atas. Harga laundry otomatis muncul dengan meng kali kan berat cucian dengan pengaturan harga perkilo yang telah kita buat sebelumnya ( $6000 \times 2$ ). Sehingga harganya Rp.12.000,-.

Sekarang yuk simak penjelasan dari proses transaksi laundry yang kita buat di atas.

Pada file transaksi\_aksi.php kita menginput sekaligus ke 2 buah tabel. Yaitu ke tabel transaksi dan tabel pakaian. Data yang kita input ke table transaksi di antaranya adalah data pelanggan yang melakukan transaksi, berat cucian, tanggal selesai, tanggal hari ini, dan status laundry nya.

```
// menangkap data yang dikirim dari form
$pelanggan = $_POST['pelanggan'];
$berat = $_POST['berat'];
$tgl_selesai = $_POST['tgl_selesai'];

$tgl_hari_ini = date('Y-m-d');
$status = 0;

// mengambil data harga per kilo dari database
$h = mysqli_query($koneksi,"select harga_per_kilo from harga");
$harga_per_kilo = mysqli_fetch_assoc($h);

// menghitung harga laundry, harga perkilo x berat cucian
$harga = $berat * $harga_per_kilo['harga_per_kilo'];

// input data ke tabel transaksi
mysqli query($koneksi,"insert into transaksi
values('','$tgl_hari_ini','$pelanggan','$harga','$berat','$tgl_selesai','$status')");
```

Untuk tanggal transaksi nya berarti kita mengambil tanggal hari ini, kemudian status laundry nya kita tetapkan dengan "0". Karena seperti yang telah di jelaskan sebelumnya, **untuk status transaksi nya 0= proses, 1= di cuci, dan 2=selesai**. Karena transaksi ini baru di input, berarti status nya kita buat 0. Artinya transaksi laundry ini masih dalam status di proses. Nanti kita juga akan buat status transaksinya ini bisa di ubah oleh admin, bisa itu di ubah status nya menjadi 1 (di cuci) atau 2 (selesai).

Oh ya, perhatikan pada bagian harga nya juga.

```
$harga = $berat * $harga_per_kilo['harga_per_kilo'];
```

Harga yang kita input ke tabel transaksi adalah berat cucian yang di input pada form input transaksi baru di kalikan dengan harga laundry perkilo dari tabel harga yang telah kita buat sebelumnya. Hasilnya kita simpan dalam variabel harga dan kita input ke tabel transaksi sekaligus dengan tanggal, pelanggan, berat, tanggal selesai dan status laundry.

```
mysqli_query($koneksi,"insert into transaksi  
values('','$tgl_hari_ini','$pelanggan','$harga','$berat','$tgl_selesai','$status')");
```

Perhatikan selanjutnya, pada saat kita selesai menginput data transaksi ke table transaksi, kita ambil id terakhir dari data yang di input pada table transaksi. Dan kita simpan dalam variabel \$id\_terakhir. Kita tangkap data yang dikirim dari form jenis\_pakaian dan jumlah\_pakaian. Karena jenis\_pakaian dan jumlah\_pakaian berbentuk array, kita tampilkan dengan menggunakan perulangan for() php.

```
// menyimpan id dari data yang di simpan pada query insert data sebelumnya  
$id_terakhir = mysqli_insert_id($koneksi);  
  
// menangkap data form input array (jenis pakaian dan jumlah pakaian)  
$jenis_pakaian = $_POST['jenis_pakaian'];  
$jumlah_pakaian = $_POST['jumlah_pakaian'];  
  
// input data cucian berdasarkan id transaksi (invoice) ke table pakaian  
for($x=0;$x<count($jenis_pakaian);$x++){  
    if($jenis_pakaian[$x] != ""){  
        mysqli_query($koneksi,"insert into pakaian
```

```
values('','$id_terakhir','$jenis_pakaian[$x]','$jumlah_pakaian[$x]')");  
}  
}  
}
```

Variabel \$x di atas menjadi urutan data array dari form jenis\_pakaian dan jumlah\_pakaian. Agar mudah di pahami bisa kita definisikan dengan data jenis\_pakaian ke 1, data jenis\_pakaian ke 2, data jenis\_pakaian ke 3 dan seterusnya sampai habis sesuai dengan jumlah form jenis\_pakaian yang ada di transaksi\_tambah.php. begitu juga dengan form array jumlah\_pakaian.

```
for($x=0;$x<count($jenis_pakaian);$x++){  
    if($jenis_pakaian[$x] != ""){  
        mysqli_query($koneksi,"insert into pakaian  
values('','$id_terakhir','$jenis_pakaian[$x]','$jumlah_pakaian[$x]')");  
    }  
}
```

Berarti di sini kita mengecek apakah data jenis\_pakaian dan jumlah\_pakaian yang ke 1 tidak kosong, jika tidak kosong (di isi) maka input ke table pakaian, kemudian berulang ke penomoran selanjutnya, yaitu 2, jika data jenis\_pakaian ke 2 tidak sama dengan kosong, jika tidak kosong (di isi) maka input ke table pakaian. Begitu terus selanjutnya sesuai dengan jumlah data jenis\_pakaian dan jumlah\_pakaian.

Oh ya pada kolumn pakaian\_transaksi, yang kita input adalah id dari data transaksi si pelanggan. Seperti pada gambar berikut yang menampilkan struktur transaksi dan pakaian.

## Membuat Edit Data Transaksi Laundry

Setelah kita selesai menampilkan data transaksi dari database dan membuat input transaksi laundry baru. Sekarang kita akan melanjutkan proses pembuatan sistem informasi laundry ini dengan membuat proses edit dan update data transaksi laundry.

Pada transaksi.php kita telah membuat tombol edit pada masing-masing record data transaksi.

```
<a href="transaksi_edit.php?id=<?php echo $d['transaksi_id']; ?>" class="btn btn-sm btn-info">Edit</a>
```

Jika di klik maka akan di alihkan ke transaksi\_edit.php sambil mengirimkan data id transaksi yang ingin di edit. Sama saja prosesnya seperti CRUD yang telah kita pelajari sebelumnya. Buat sebuah file baru dalam folder admin dengan nama "*transaksi\_edit.php*". pada transaksi\_edit.php ini kita akan membuat form untuk mengedit data transaksi.

### transaksi\_edit.php

```
<?php include 'header.php'; ?>

<?php

// koneksi database
include '../koneksi.php';
?>
<div class="container">
    <div class="panel">
        <div class="panel-heading">
            <h4>Edit Transaksi Laundry</h4>
        </div>
        <div class="panel-body">

            <div class="col-md-8 col-md-offset-2">
                <a href="transaksi.php" class="btn btn-sm btn-info pull-right">Kembali</a>
                <br/>
                <br/>
                <?php
                    // menangkap id yang dikirim melalui url
                    $id = $_GET['id'];

                    // megambil data pelanggan yang ber id di atas dari
                    tabel pelanggan
                    $transaksi = mysqli_query($koneksi, "select * from
transaksi where transaksi_id='$id'");
                    while($t=mysqli_fetch_array($transaksi)){
                        ?
                        <form method="post"
action="transaksi_update.php">
                            <!-- menyimpan id transaksi yang di edit
dalam form hidden berikut -->
                            <input type="hidden" name="id"
value="<?php echo $t['transaksi_id']; ?>">
```

```

        <div class="form-group">
            <label>Pelanggan</label>
            <select class="form-control"
name="pelanggan" required="required">
                <option value="">- Pilih
                Pelanggan</option>
                <?php
                    // mengambil data pelanggan
                    dari database
                    $data =
                    mysqli_query($koneksi,"select * from pelanggan");
                    // mengubah data ke array dan
                    menampilkannya dengan perulangan while
                    while($d=mysqli_fetch_array($data)){
                        ?>
                        <option <?php
                        if($d['pelanggan_id']==$t['transaksi_pelanggan']){
                            echo
                            "selected='selected'"; } ?> value=<?php echo $d['pelanggan_id']; ?>><?php
                            echo $d['pelanggan_nama']; ?></option>
                            <?php
                        }
                        ?>
                    </select>
                </div>

                <div class="form-group">
                    <label>Berat</label>
                    <input type="number" class="form-
control" name="berat" placeholder="Masukkan berat cucian .."
required="required" value=<?php echo $t['transaksi_berat']; ?>>
                </div>

                <div class="form-group">
                    <label>Tgl. Selesai</label>
                    <input type="date" class="form-
control" name="tgl_selesai" required="required" value=<?php echo
$t['transaksi_tgl_selesai']; ?>>
                </div>

                <br/>

                <table class="table table-bordered table-
striped">
                    <tr>
                        <th>Jenis Pakaian</th>
                        <th width="20%">Jumlah</th>
                    </tr>
                    <?php
                        // menyimpan id transaksi ke
                        variabel id_transaksi

```

```

        $id_transaksi = $t['transaksi_id'];

        // menampilkan pakaian-pakaian dari
transaksi yang ber id di atas
        $pakaian =
mysqli_query($koneksi,"select * from pakaian where
pakaian_transaksi='{$id_transaksi}'");

while($p=mysqli_fetch_array($pakaian)){
    ?>
<tr>
    <td><input type="text"
class="form-control" name="jenis_pakaian[]" value="php echo
$p['pakaian_jenis']; ?"></td>
    <td><input
type="number" class="form-control" name="jumlah_pakaian[]" value="php echo
$p['pakaian_jumlah']; ?"></td>
</tr>
<?php } ?>
<tr>
    <td><input type="text"
class="form-control" name="jenis_pakaian[]"></td>
    <td><input
type="number" class="form-control" name="jumlah_pakaian[]"></td>
</tr>
<tr>
    <td><input type="text"
class="form-control" name="jenis_pakaian[]"></td>
    <td><input
type="number" class="form-control" name="jumlah_pakaian[]"></td>
</tr>
<tr>
    <td><input type="text"
class="form-control" name="jenis_pakaian[]"></td>
    <td><input
type="number" class="form-control" name="jumlah_pakaian[]"></td>
</tr>
<tr>
    <td><input type="text"
class="form-control" name="jenis_pakaian[]"></td>
    <td><input
type="number" class="form-control" name="jumlah_pakaian[]"></td>
</tr>
</table>

<div class="form-group alert alert-
info">
    <label>Status</label>
    <select class="form-control"
name="status" required="required">
        <option <?php
if($t['transaksi_status']=="0") {echo "selected='selected'"; } ?>
value="0">PROSES</option>

```

```

        <option <?php
if($t['transaksi_status']=="1") {echo "selected='selected'"; } ?> value="1">DI
CUCI</option>

        <option <?php
if($t['transaksi_status']=="2") {echo "selected='selected'"; } ?>
value="2">SELESAI</option>

                </select>
            </div>

            <input type="submit" class="btn
btn-primary" value="Simpan">
        </form>
    <?php
}
?>
</div>

        </div>
    </div>
</div>

<?php include 'footer.php'; ?>

```

## Sekarang coba klik edit pada data transaksi

The screenshot shows a web application interface for managing laundry transactions. At the top, there's a navigation bar with links like 'Dashboard', 'Pelanggan', 'Transaksi', 'Laporan', 'Pengaturan', and 'Log Out'. The main content area is titled 'Data Transaksi Laundry'. Below the title is a table with columns: No, Invoice, Tanggal, Pelanggan, Berat (Kg), Tgl. Selesai, Harga, Status, and OPSI. There are two rows of data. The second row, which corresponds to the transaction ID 1 (INVOICE-1), has an 'Edit' button highlighted with a red box and a red arrow pointing to it from below.

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	OPSI
1	INVOICE-1	2018-04-06	Sumanto	2	2018-04-09	Rp. 12,000,-	PROSES	<a href="#">Invoice</a> <a href="#">Edit</a> <a href="#">Batalkan</a>

Maka akan muncul form untuk edit data transaksi laundry.

LAUNDRY Malas Ngoding

Dashboard Pelanggan Transaksi Laporan Pengaturan Log Out

Halo, admin !

Edit Transaksi Laundry

Pelanggan: Sumantri

Berat: 2

Tgl. Selesai: 04 / 09 / 2018

Jenis Pakaian	Jumlah
Celana Jeans	2
Baju Kemeja	1
Baju Kaos	4

Data transaksi yang ingin di edit pun muncul pada form edit data transaksi. Seperti yang bisa teman-teman lihat. Pada form edit transaksi ini kita telah menetapkan file yang akan memproses form ini adalah file `transaksi_update.php`.

```
<form method="post" action="transaksi_update.php">
```

Oleh karena itu buat sebuah file baru lagi dalam folder admin dengan nama `transaksi_update.php`

`transaksi_update.php`

```
<?php
// menghubungkan dengan koneksi
include '../koneksi.php';

// menangkap data yang dikirim dari form
$id = $_POST['id'];
$pelanggan = $_POST['pelanggan'];
$berat = $_POST['berat'];
$tgl_selesai = $_POST['tgl_selesai'];

$status = $_POST['status'];

// mengambil data harga per kilo dari database
$h = mysqli_query($koneksi, "select harga_per_kilo from harga");
$harga_per_kilo = mysqli_fetch_assoc($h);

// menghitung harga laundry, harga perkilo x berat cucian
$harga = $berat * $harga_per_kilo['harga_per_kilo'];
```

```

// update data transaksi
mysqli_query($koneksi,"update transaksi set transaksi_pelanggan='$pelanggan',
transaksi_harga='$harga', transaksi_berat='$berat',
transaksi_tgl_selesai='$tgl_selesai', transaksi_status='$status' where
transaksi_id='$id')";

// menangkap data form input array (jenis pakaian dan jumlah pakaian)
$jenis_pakaian = $_POST['jenis_pakaian'];
$jumlah_pakaian = $_POST['jumlah_pakaian'];

// menghapus semua pakaian dalam transaksi ini
mysqli_query($koneksi,"delete from pakaian where pakaian_transaksi='$id')";

// input ulang data cucian berdasarkan id transaksi (invoice) ke table
pakaian
for($x=0;$x<count($jenis_pakaian);$x++) {
    if($jenis_pakaian[$x] != "") {
        mysqli_query($koneksi,"insert into pakaian
values('','$id','$jenis_pakaian[$x]','$jumlah_pakaian[$x]')");
    }
}

header("location:transaksi.php");

?>

```

Pada file transaksi\_update.php di atas, tidak jauh berbeda dengan aksi yang terdapat dalam file transaksi\_aksi.php. pada transaksi\_update.php di atas juga sudah saya tuliskan comment untuk masing-masing penjelasan dari syntaxnya.

Sekarang coba ubah data transaksi sesuai keinginan teman-teman. Di sini saya mencoba merubah data nya sesuai gambar berikut.

Edit Transaksi Laundry

Pelanggan: Sumantri

Berat: 3

Tgl. Selesai: 04 / 08 / 2018

Jenis Pakaian	Jumlah
Celana Jeans	2
Baju Kemeja	1
Baju Kaos	4
Celana Pendek	1

Saya mengubah berat cuciannya menjadi 3. Dan tanggal selesai nya menjadi tanggal 8 april. Dan saya juga menambahkan cucian si pelanggan. Yaitu menambah celana pendek sejumlah 1 buah.

Berat: 3

Tgl. Selesai: 04 / 08 / 2018

Jenis Pakaian	Jumlah
Celana Jeans	2
Baju Kemeja	1
Baju Kaos	4
Celana Pendek	1

Status: DI CUCI

**Simpan**

Disini saya juga mencoba merubah status transaksi ini dari PROSES menjadi DI CUCI. Dan klik simpan. Ini berarti status transaksi ini telah selesai di cuci. Dan menunggu cucian di ambil oleh pelanggan. Dan kemudian kita bisa merubah statusnya lagi menjadi SELESAI.

Setelah di ubah, klik simpan.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/transaksi.php`. The page title is "Data Transaksi Laundry". A table displays a single transaction row:

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	Opsi
1	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI	<button>Invoice</button> <button>Edit</button> <button>Batalkan</button>

The "Berat (Kg)" field (containing "3") and the "Status" field (containing "DICUCI") are highlighted with red boxes. There is a blue button labeled "Transaksi Baru" at the top right of the table area.

Nah, sampai di sini data berhasil di ubah. Dari yang tadi nya berat nya adalah 2, berubah menjadi 3. Sehingga harga laundry akan otomatis di hitung ulang dengan meng kali kan harga laundry perkilo dengan berat cucian ( $6000 \times 3$ ). Begitu juga dengan tanggal selesai dan status transaksi.

Saya juga mencoba menambah lagi data transaksi nya dengan menginput beberapa data percobaan.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/transaksi.php`. The page title is "Data Transaksi Laundry". A table displays two transaction rows:

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	Opsi
1	INVOICE-2	2018-04-06	Sri Dayatun	2	2018-04-09	Rp. 12,000,-	PROSES	<button>Invoice</button> <button>Edit</button> <button>Batalkan</button>
2	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI	<button>Invoice</button> <button>Edit</button> <button>Batalkan</button>

The "Berat (Kg)" field for the second row (containing "3") and the "Status" field for the first row (containing "PROSES") are highlighted with red boxes. There is a blue button labeled "Transaksi Baru" at the top right of the table area.

Selesai sudah proses edit transaksi yang kita buat. Selanjutnya akan kita buat cetak invoice transaksi untuk bisa di berikan ke pelanggan sebagai bukti laundry.

## Membuat Invoice Transaksi Laundry

Perhatikan pada tombol invoice di transaksi.php. kita telah membuat link mengarahkan halaman ke halaman transaksi\_invoice.php.

```
<a href="transaksi_invoice.php?id=<?php echo $d['transaksi_id']; ?>"  
target="_blank" class="btn btn-sm btn-warning">Invoice</a>
```

Pada file transaksi\_invoice.php ini kita akan membuat invoice per transaksinya. Seperti yang teman-teman bisa lihat pada link tujuan pada tombol Invoice di atas. Kita juga mengirimkan parameter id melalui url. Sama seperti pada tombol edit yang telah kita buat sebelumnya.

Buat sebuah file baru dengan nama “transaksi\_invoice.php” dalam folder admin.

transaksi\_invoice.php

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Sistem Informasi Laundry Malas Ngoding -  
WWW.MALASNGODING.COM</title>  
  
    <link rel="stylesheet" type="text/css"  
href="../assets/css/bootstrap.css">  
    <script type="text/javascript" src="../assets/js/jquery.js"></script>  
    <script type="text/javascript"  
src="../assets/js/bootstrap.js"></script>  
  
</head>  
<body>  
    <!-- cek apakah sudah login -->  
    <?php  
    session_start();  
    if($_SESSION['status']!="login") {  
        header("location:../index.php?pesan=belum_login");  
    }  
    ?>  
  
    <?php  
    // koneksi database  
    include '../koneksi.php';  
    ?>  
    <div class="container">
```

```

<div class="col-md-10 col-md-offset-1">

    <?php
    // menangkap id yang dikirim melalui url
    $id = $_GET['id'];

    // megambil data pelanggan yang ber id di atas dari tabel
pelanggan
    $transaksi = mysqli_query($koneksi,"select * from
transaksi,pelanggan where transaksi_id='$id' and
transaksi pelanggan=pelanggan id");
    while($t=mysqli_fetch_array($transaksi)){
        ?
        <center><h2>LAUNDRY " Malas Ngoding "</h2></center>
        <h3></h3>

        <a href="transaksi_invoice_cetak.php?id=<?php echo
$id; ?>" target="_blank" class="btn btn-primary pull-right"><i
class="glyphicon glyphicon-print"></i> CETAK</a>

        <br/>
        <br/>

        <table class="table">
            <tr>
                <th width="20%">No. Invoice</th>
                <th>:</th>
                <td>INVOICE-<?php echo
$t['transaksi_id']; ?></td>
            </tr>
            <tr>
                <th width="20%">Tgl. Laundry</th>
                <th>:</th>
                <td><?php echo $t['transaksi_tgl'];
?></td>
            </tr>
            <tr>
                <th>Nama Pelanggan</th>
                <th>:</th>
                <td><?php echo $t['pelanggan_nama'];
?></td>
            </tr>
            <tr>
                <th>HP</th>
                <th>:</th>
                <td><?php echo $t['pelanggan_hp'];
?></td>
            </tr>
            <tr>
                <th>Alamat</th>
                <th>:</th>
                <td><?php echo $t['pelanggan_alamat'];
?></td>
            </tr>
            <tr>

```

```

        <th>Berat Cucian (Kg)</th>
        <th>:</th>
        <td><?php echo $t['transaksi_berat'];
?></td>
    </tr>
    <tr>
        <th>Tgl. Selesai</th>
        <th>:</th>
        <td><?php echo
$t['transaksi_tgl_selesai']; ?></td>
    </tr>
    <tr>
        <th>Status</th>
        <th>:</th>
        <td>
            <?php
            if($t['transaksi_status']=="0") {
                echo "<div class='label
label-warning'>PROSES</div>";
            } else
            if($t['transaksi_status']=="1") {
                echo "<div class='label
label-info'>DI CUCI</div>";
            } else
            if($t['transaksi_status']=="2") {
                echo "<div class='label
label-success'>SELESAI</div>";
            }
        ?>
    </td>
    </tr>
    <tr>
        <th>Harga</th>
        <th>:</th>
        <td><?php echo "Rp.
".number_format($t['transaksi_harga'])." , -"; ?></td>
    </tr>
</table>

<br/>

<h4 class="text-center">Daftar Cucian</h4>
<table class="table table-bordered table-striped">
    <tr>
        <th>Jenis Pakaian</th>
        <th width="20%">Jumlah</th>
    </tr>

    <?php
    // menyimpan id transaksi ke variabel
    id_transaksi
    $id = $t['transaksi_id'];

    // menampilkan pakaian-pakaian dari transaksi
    yang ber id di atas
    $pakaian = mysqli_query($koneksi,"select * from
pakaian where pakaian_transaksi='$id'");

```

```

        while($p=mysqli_fetch_array($pakaian)) {
            ?
            <tr>
                <td><?php echo $p['pakaian_jenis'];
?></td>
$p['pakaian_jumlah']; ?></td>
            </tr>
            <?php } ?>

        </table>

        <br/>
        <p><center><i>" Terima kasih telah
mempercayakan cucian anda pada kami ".</i></center></p>

        <?php
    }
    ?>
</div>

</div>

</body>
</html>

```

Tidak begitu sulit. Intinya di sini kita menampilkan semua data transaksi yang ber id sesuai dengan yang ingin di cetak invoice nya.

Coba klik pada tombol invoice.

No. Invoice	:	INVOICE-1
Tgl. Laundry	:	2018-04-06
Nama Pelanggan	:	Sumantri
HP	:	08123456789
Alamat	:	Jl. cenderawasih no.126 bandung
Berat Cucian (Kg)	:	3
Tgl. Selesai	:	2018-04-08
Status	:	DILAKUKAN
Harga	:	Rp. 18,000,-

Daftar Cucian

Jenis Pakaian	Jumlah
Celana Jeans	2
Baju Kemeja	1
Baju Kaos	4
Celana Pendek	1

Nah, invoice pun selesai. Perlu teman-teman perhatikan disini, untuk nomor invoicenya kita membuat penomorannya dari id transaksi.

```
<tr>
    <th width="20%">No. Invoice</th>
    <th>:</th>
    <td>INVOICE-<?php echo $t['transaksi_id']; ?></td>
</tr>
```

Selesai. Selanjutnya kita akan mengaktifkan tombol cetak pada invoice agar invoice tersebut bisa dicetak langsung untuk pelanggan.

## Membuat Cetak Invoice Transaksi Laundry

Pada invoice transaksi kita telah membuat tombol cetak. Kita mau saat tombol save di klik, maka invoice akan di cetak.

LAUNDRY " Malas Ngoding "

No. Invoice	:	INVOICE-2
Tgl. Laundry	:	2018-04-06
Nama Pelanggan	:	Sri Dayatun
HP	:	08736363444
Alamat	:	Jl. tangkuban perahu no.23 Bandung
Berat Cucian (Kg)	:	2
Tgl. Selesai	:	2018-04-09
Status	:	<b>PROSES</b>
Harga	:	Rp. 12,000,-

Daftar Cucian

Jenis Pakaian	Jumlah
Baju Pima	2
Selimut	1

\* Terima kasih telah mempercayakan cucian anda pada kami \*.

Pada tombol cetak di atas sudah kita buat link nya yang menuju ke

```
<a href="transaksi_invoice_cetak.php?id=<?php echo $id; ?>" target="_blank"
class="btn btn-primary pull-right"><i class="glyphicon glyphicon-print"></i>
CETAK</a>
```

Di alihkan ke halaman transaksi\_invoice\_cetak.php sambil mengirim id transaksi.

Buat sebuah file baru dalam folder admin dengan nama “transaksi\_invoice\_cetak.php”.

transaksi\_invoice\_cetak.php

```
<!DOCTYPE html>
<html>
<head>
<title>Sistem Informasi Laundry Malas Ngoding - WWW.MALASNGODING.COM</title>

<link rel="stylesheet" type="text/css" href="../assets/css/bootstrap.css">
<script type="text/javascript" src="../assets/js/jquery.js"></script>
<script type="text/javascript" src="../assets/js/bootstrap.js"></script>

</head>
<body>
<!-- cek apakah sudah login -->
<?php
session_start();
if($_SESSION['status']!='login') {
    header("location:../index.php?pesan=belum_login");
}
?>
```

```

<?php
// koneksi database
include '../koneksi.php';
?>
<div class="container">

    <div class="col-md-10 col-md-offset-1">

        <?php
        // menangkap id yang dikirim melalui url
        $id = $_GET['id'];

        // megambil data pelanggan yang ber id di atas dari tabel
pelanggan
        $transaksi = mysqli_query($koneksi, "select * from
transaksi,pelanggan where transaksi_id='$id' and
transaksi_pelanggan=pelanggan_id");
        while($t=mysqli_fetch_array($transaksi)) {
            ?>
            <center><h2>LAUNDRY " Malas Ngoding "</h2></center>
            <h3>INVOICE-<?php echo $t['transaksi_id']; ?></h3>

            <br/>

            <table class="table">
                <tr>
                    <th width="20%">Tgl. Laundry</th>
                    <th>:</th>
                    <td><?php echo $t['transaksi_tgl']; ?></td>
                </tr>
                <tr>
                    <th>Nama Pelanggan</th>
                    <th>:</th>
                    <td><?php echo $t['pelanggan_nama']; ?></td>
                </tr>
                <tr>
                    <th>HP</th>
                    <th>:</th>
                    <td><?php echo $t['pelanggan_hp']; ?></td>
                </tr>
                <tr>
                    <th>Alamat</th>
                    <th>:</th>
                    <td><?php echo $t['pelanggan_alamat']; ?></td>
                </tr>
                <tr>
                    <th>Berat Cucian (Kg)</th>
                    <th>:</th>
                    <td><?php echo $t['transaksi_berat']; ?></td>
                </tr>
                <tr>
                    <th>Tgl. Selesai</th>
                    <th>:</th>
                    <td><?php echo $t['transaksi_tgl_selesai']; ?></td>
                </tr>
            </table>
        ?></td>
    </div>

```

```

</tr>
<tr>
    <th>Status</th>
    <th>:</th>
    <td>
        <?php
        if($t['transaksi_status']=="0") {
            echo "<div class='label label-
warning'>PROSES</div>";
        } else if($t['transaksi_status']=="1") {
            echo "<div class='label label-
info'>DI CUCI</div>";
        } else if($t['transaksi_status']=="2") {
            echo "<div class='label label-
success'>SELESAI</div>";
        }
        ?>
    </td>
</tr>
<tr>
    <th>Harga</th>
    <th>:</th>
    <td><?php echo "Rp.
".number_format($t['transaksi_harga'])." ,-" ; ?></td>
</tr>
</table>

<br/>

<h4>Daftar Cucian</h4>
<table class="table table-bordered table-striped">
    <tr>
        <th>Jenis Pakaian</th>
        <th width="20%">Jumlah</th>
    </tr>

    <?php
    // menyimpan id transaksi ke variabel id_transaksi
    $id = $t['transaksi_id'];

    // menampilkan pakaian-pakaian dari transaksi yang
ber id di atas
    $pakaian = mysqli_query($koneksi,"select * from
pakaian where pakaian_transaksi='$id'");

    while($p=mysqli_fetch_array($pakaian)){
        ?>
        <tr>
            <td><?php echo $p['pakaian_jenis'];
?></td>
            <td width="5%"><?php echo
$p['pakaian_jumlah']; ?></td>
        </tr>
        <?php } ?>
    </table>

<br/>

```

```

<p><center><i>" Terima kasih telah mempercayakan
cucian anda pada kami ".</i></center></p>

<?php
}
?>
</div>

</div>

<script type="text/javascript">
    window.print();
</script>

</body>
</html>

```

Isinya sama saja dengan file transaksi\_invoice.php. bedanya hanya pada file transaksi\_invoice\_cetak.php kita menambahkan syntax berikut untuk membuat perintah print.

```

<script type="text/javascript">
    window.print();
</script>

```

Sekarang coba klik pada tombol cetak.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/transaksi_invoice_cetak.php?id=2`. The page title is "LAUNDRY " Malas Ngoding ". The main content displays an invoice for laundry services. On the left, a "Print" dialog box is open, showing settings for "Microsoft Print to PDF". The invoice details include:

- Date: 2018-04-06
- Name: Sri Dayatun
- Phone: 08736363444
- Address: jl. tangkuban perahu no.23 Bandung
- Quantity: 2
- Date: 2018-04-09
- Status: PROSES
- Total: Rp. 12,000,-

Below the invoice, there is a table titled "Daftar Cucian" (Laundry List) showing:

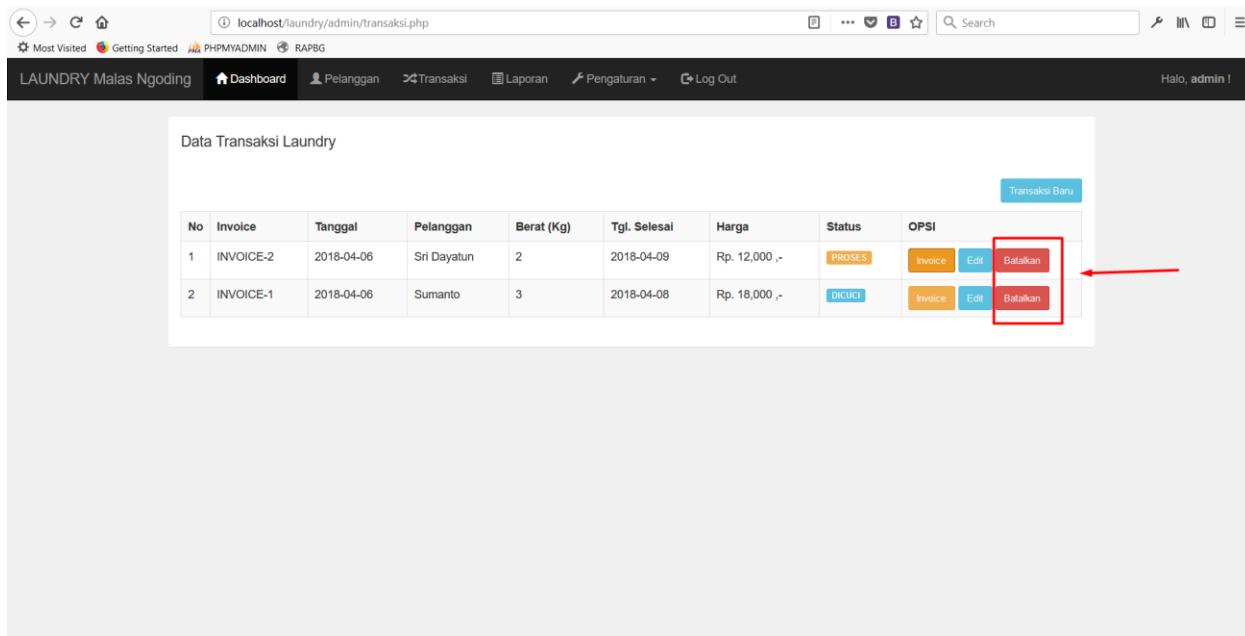
Jenis Pakalan	Jumlah
Baju Plama	2
Selimut	1

A small note at the bottom states: "Terima kasih telah mempercayakan cucian anda pada kami".

Oke selesai.

## Membuat Batalkan Transaksi Laundry

Sampai lah kita pada tahap trakhir dalam membuat transaksi laundry. Kita akan membuat proses membatalkan transaksi laundry. Pada transaksi.php kita telah membuat tombol batalkan.



The screenshot shows a web-based laundry management system. At the top, there's a navigation bar with links like 'localhost/laundry/admin/transaksi.php', 'Dashboard', 'Pelanggan', 'Transaksi', 'Laporan', 'Pengaturan', and 'Log Out'. The main content area is titled 'Data Transaksi Laundry'. It displays a table with two rows of transaction data:

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	OPSI
1	INVOICE-2	2018-04-06	Sri Dayatun	2	2018-04-09	Rp. 12,000,-	PROSES	<a href="#">Invoice</a> <a href="#">Edt</a> <a href="#">Batalkan</a>
2	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI	<a href="#">Invoice</a> <a href="#">Edt</a> <a href="#">Batalkan</a>

A red box highlights the 'Batalkan' button for the second transaction (INVOICE-1). A red arrow points from the text above to this button.

Jika tombol atau link batalkan di atas di klik, maka halaman akan dialihkan ke "transaksi\_hapus.php".

```
<a href="transaksi_hapus.php?id=<?php echo $d['transaksi_id']; ?>" class="btn btn-sm btn-danger">Batalkan</a>
```

Buat sebuah file baru dalam folder admin dengan nama "transaksi\_hapus.php".

`transaksi_hapus.php`

```
<?php
// menghubungkan koneksi
include '../koneksi.php';

// menangkap data id yang dikirim dari url
$id = $_GET['id'];

// menghapus transaksi
mysqli_query($koneksi,"delete from transaksi where transaksi id='$id'");
```

```

// menghapus data pakaian dalam transaksi ini
mysqli_query($koneksi,"delete from pakaian where pakaian_transaksi='$id')";

// alihkan halaman ke halaman transaksi
header("location:transaksi.php");
?>

```

Proses penghapusan data transaksi sama saja seperti proses CRUD yang telah kita pelajari sebelumnya. Hanya saja di sini tidak hanya data transaksi yang kita hapus. Tapi data pakaian yang memiliki id transaksi yang sama akan dihapus juga.

```

// menghapus data pakaian dalam transaksi ini
mysqli_query($koneksi,"delete from pakaian where pakaian_transaksi='$id')";

```

Sekarang coba klik salah satu tombol batalkan, tergantung data transaksi yang mana yang ingin teman-teman hapus.

Di sini saya akan mencoba menghapus transaksi "sri dayatun".

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	OPSI
1	INVOICE-2	2018-04-06	Sri Dayatun	2	2018-04-09	Rp. 12,000,-	<span style="background-color: orange;">PROSES</span>	<span style="background-color: orange;">Invoice</span> <span style="background-color: blue;">Edit</span> <span style="background-color: red; border: 2px solid red; padding: 2px;">Batalkan</span>
2	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	<span style="background-color: blue;">DICUCI</span>	<span style="background-color: orange;">Invoice</span> <span style="background-color: blue;">Edit</span> <span style="background-color: red; border: 2px solid red; padding: 2px;">Batalkan</span>

Maka data transaksi akan dihapus.

The screenshot shows a web-based laundry management system. At the top, there's a navigation bar with links like 'Dashboard', 'Pelanggan', 'Transaksi', 'Laporan', 'Pengaturan', and 'Log Out'. A search bar is also present. The main content area is titled 'Data Transaksi Laundry' and displays a table of transactions. The table has columns for No, Invoice, Tanggal, Pelanggan, Berat (Kg), Tgl. Selesai, Harga, Status, and Opsi. A single row is shown with values: No 1, Invoice INVOICE-1, Tanggal 2018-04-06, Pelanggan Sumarto, Berat (Kg) 3, Tgl. Selesai 2018-04-08, Harga Rp. 18,000,-, Status DILAKUKAN, and Opsi buttons for 'DIBATALKAN', 'Invoice', 'Edit', and 'Batalkan'. A red arrow points to the 'Batalkan' button.

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status	Opsi
1	INVOICE-1	2018-04-06	Sumarto	3	2018-04-08	Rp. 18,000,-	DILAKUKAN	<a href="#">Invoice</a> <a href="#">Edit</a> <a href="#">Batalkan</a>

Selesai sudah proses batalkan transaksi. Sampai di sini berarti kita telah selesai membuat semua proses transaksi pada sistem informasi laundry ini. Selanjutnya kita akan membuat filter laporan transaksi dan cetak data laporan transaksi.

# Membuat Laporan Laundry

Membuat laporan transaksi laundry merupakan fitur terakhir yang kita buat. Fitur laporan ini berguna untuk bisa memfilter data laporan. Misalnya admin ingin menampilkan data transaksi perhari, perbulan dan pertahun. Dengan cara memfilter dari tanggal berapa sampai tanggal berapa data transaksi yang ingin di tampilkan.

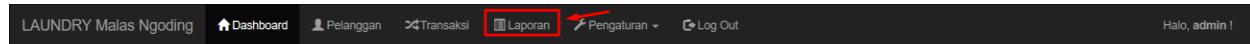
Setelah data transaksi laundry di tampilkan sesuai dengan rentang tanggal yang sudah di tentukan. Maka kita juga akan membuat 2 buah tombol cetak laporan. Yaitu cetak laporan berupa print. Dan cetak laporan berupa file PDF.

Langsung saja kita mulai.

## Membuat Filter Laporan

Silahkan teman-teman perhatikan lagi pada link menu laporan. Pada menu laporan kita telah membuat link yang mengarah ke file laporan.php.

```
<li><a href="laporan.php"><i class="glyphicon glyphicon-list-alt"></i>  
Laporan</a></li>
```



Buat sebuah file php baru dalam folder admin dengan nama laporan.php

laporan.php.

```
<?php include 'header.php'; ?>

<div class="container">
    <div class="panel">
        <div class="panel-heading">
            <h4>Filter Laporan</h4>
        </div>
        <div class="panel-body">

            <form action="laporan.php" method="get">
                <table class="table table-bordered table-striped">
                    <tr>
                        <th>Dari Tanggal</th>
                        <th>Sampai Tanggal</th>
                        <th width="1%"></th>
                    </tr>
                    <tr>
```

```

        <td>
            <br/>
            <input type="date" name="tgl_dari"
class="form-control">
        </td>
        <td>
            <br/>
            <input type="date"
name="tgl_sampai" class="form-control">
        </td>
        <td>
            <br/>
            <input type="submit" class="btn
btn-primary" value="Filter">
        </td>
    </tr>

    </table>
</form>

</div>
</div>

<br/>

<?php
if(isset($_GET['tgl_dari']) && isset($_GET['tgl_sampai'])){

    $dari = $_GET['tgl_dari'];
    $sampai = $_GET['tgl_sampai'];

    ?>
    <div class="panel">
        <div class="panel-heading">
            <h4>Data Laporan Laundry dari <b><?php echo $dari;
?></b> sampai <b><?php echo $sampai; ?></b></h4>
        </div>
        <div class="panel-body">

            <a target="_blank" href="cetak_print.php?dari=<?php
echo $dari; ?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-
primary"><i class="glyphicon glyphicon-print"></i> CETAK</a>
            <a target="_blank" href="cetak_pdf.php?dari=<?php
echo $dari; ?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-
primary"><i class="glyphicon glyphicon-print"></i> CETAK PDF</a>
            <br/>
            <br/>
            <table class="table table-bordered table-striped">
                <tr>
                    <th width="1%">No</th>
                    <th>Invoice</th>
                    <th>Tanggal</th>
                    <th>Pelanggan</th>
                    <th>Berat (Kg)</th>
                    <th>Tgl. Selesai</th>
                    <th>Harga</th>
                </tr>

```

```

<th>Status</th>
</tr>

<?php
// koneksi database
include '../koneksi.php';

// mengambil data pelanggan dari database
$data = mysqli_query($koneksi,"select * from
pelanggan,transaksi where transaksi_pelanggan=pelanggan_id and
date(transaksi_tgl) > '$dari' and date(transaksi_tgl) < '$sampai' order by
transaksi_id desc");
$no = 1;
// mengubah data ke array dan menampilkannya
dengan perulangan while
while($d=mysqli_fetch_array($data)){
?
<tr>
<td><?php echo $no++; ?></td>
<td>INVOICE-<?php echo
$d['transaksi_id']; ?></td>
<td><?php echo $d['transaksi_tgl'];
<td><?php echo
$d['pelanggan_nama']; ?></td>
<td><?php echo
$d['transaksi_berat']; ?></td>
<td><?php echo
$d['transaksi_tgl_selesai']; ?></td>
<td><?php echo "Rp.
".number_format($d['transaksi_harga'])."-"; ?></td>
<td>
<?php
if($d['transaksi_status']=="0") {
echo "<div class='label
label-warning'>PROSES</div>";
} else
if($d['transaksi_status']=="1") {
echo "<div class='label
label-info'>DICUCI</div>";
} else
if($d['transaksi_status']=="2") {
echo "<div class='label
label-success'>SELESAI</div>";
}
?>
</td>
</tr>
<?php
}
?>

```

```

        </table>
    </div>
</div>
<?php } ?>

</div>

<?php include 'footer.php'; ?>

```

Pada file laporan.php kita membuat sebuah form penginputan tanggal.

```

<form action="laporan.php" method="get">
    <table class="table table-bordered table-striped">
        <tr>
            <th>Dari Tanggal</th>
            <th>Sampai Tanggal</th>
            <th width="1%"></th>
        </tr>
        <tr>
            <td>
                <br/>
                <input type="date" name="tgl_dari" class="form-control">
            </td>
            <td>
                <br/>
                <input type="date" name="tgl_sampai" class="form-control">
            </td>
            <td>
                <br/>
                <input type="submit" class="btn btn-primary"
value="Filter">
            </td>
        </tr>
    </table>
</form>

```

Yaitu form tanggal dari dan tanggal sampai. Maksudnya untuk menentukan dari tanggal berapa sampai tanggal berapa data akan di tampilkan.

Pada form ini, kita menetapkan file aksinya di file yang sama. Yaitu di file laporan.php juga. Dengan menggunakan method GET.

```
<form action="laporan.php" method="get">
```

Sebenarnya pada action bisa juga di kosongkan agar form ini di proses pada file yang sama. Tapi di sini kita buat saja.

Selanjutnya perhatikan lagi di bawah form.

```
<?php
if(isset($_GET['tgl_dari']) && isset($_GET['tgl_sampai'])){

    $dari = $_GET['tgl_dari'];
    $sampai = $_GET['tgl_sampai'];

?>
<div class="panel">
    <div class="panel-heading">
        <h4>Data Laporan Laundry dari <b><?php echo $dari; ?></b>
    sampai <b><?php echo $sampai; ?></b></h4>
    </div>
    <div class="panel-body">

        <a target="_blank" href="cetak_print.php?dari=<?php echo $dari;
?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-primary"><i
class="glyphicon glyphicon-print"></i> CETAK</a>
        <a target="_blank" href="cetak_pdf.php?dari=<?php echo $dari;
?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-primary"><i
class="glyphicon glyphicon-print"></i> CETAK PDF</a>
        <br/>
        <br/>
        <table class="table table-bordered table-striped">
            <tr>
                <th width="1%">No</th>
                <th>Invoice</th>
                <th>Tanggal</th>
                <th>Pelanggan</th>
                <th>Berat (Kg)</th>
                <th>Tgl. Selesai</th>
                <th>Harga</th>
                <th>Status</th>
            </tr>
            <?php
            // koneksi database
            include '../koneksi.php';

            // mengambil data pelanggan dari database
            $data = mysqli_query($koneksi,"select * from pelanggan,transaksi
            where transaksi_pelanggan=pelanggan_id and date(transaksi_tgl) > '$dari' and
            date(transaksi_tgl) < '$sampai' order by transaksi_id desc");
            $no = 1;
            // mengubah data ke array dan menampilkannya dengan perulangan
while
            while($d=mysqli_fetch_array($data)) {
                ?>
                <tr>
                    <td><?php echo $no++; ?></td>
                    <td>INVOICE-<?php echo $d['transaksi_id']; ?></td>
                    <td><?php echo $d['transaksi_tgl']; ?></td>
```

```

<td><?php echo $d['pelanggan_nama']; ?></td>
<td><?php echo $d['transaksi_berat']; ?></td>
<td><?php echo $d['transaksi_tgl_selesai']; ?></td>
<td><?php echo "Rp.
".number_format($d['transaksi_harga'])." , -"; ?></td>
<td>
<?php

if($d['transaksi_status']=="0") {
    echo "<div class='label label-warning'>PROSES</div>";
} else if($d['transaksi_status']=="1") {
    echo "<div class='label label-info'>DICUCI</div>";
} else if($d['transaksi_status']=="2") {
    echo "<div class='label label-success'>SELESAI</div>";
}
?>
</td>
</tr>
<?php
}
?>
</table>
</div>
</div>
<?php } ?>
```

Di sini kita mengecek apakah ada data tanggal yang di input. Jika ada maka tampilkan data transaksi nya berdasarkan tanggal yang ada (data tanggal yang di kirim menggunakan GET melalui URL).

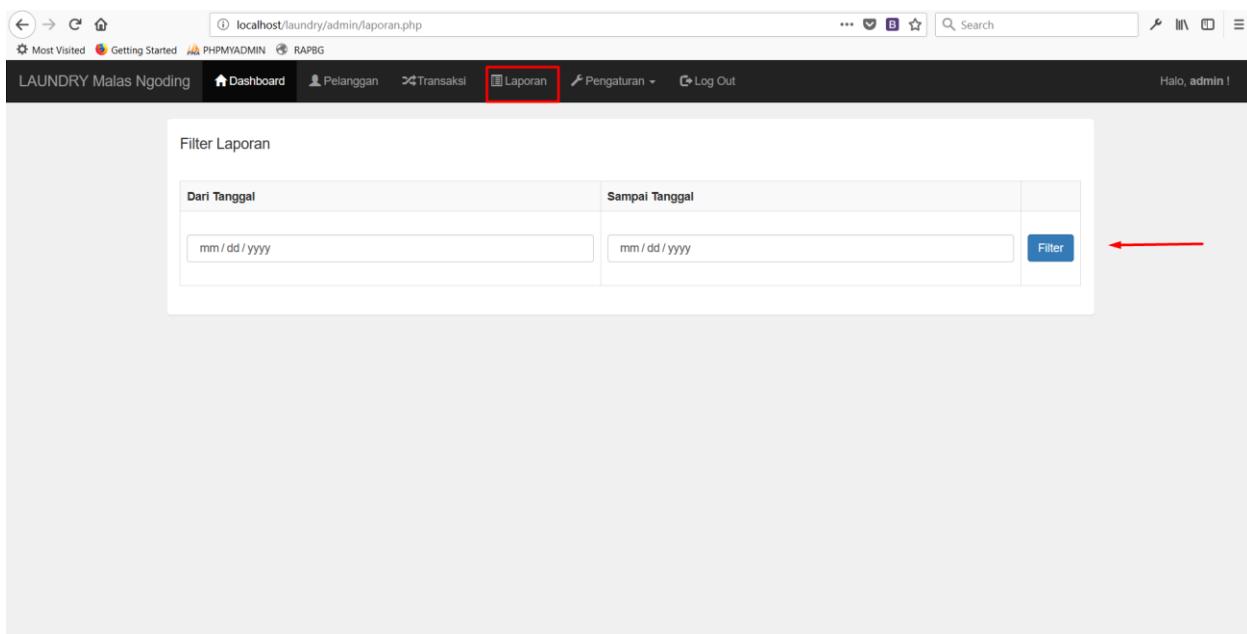
Sedikit saya jelaskan lagi tentang query untuk mengambil data berdasarkan tanggal seperti pada syntax di atas.

```
$data = mysqli_query($koneksi,"select * from pelanggan,transaksi where
transaksi_pelanggan=pelanggan_id and date(transaksi_tgl) > '$dari' and
date(transaksi_tgl) < '$sampai' order by transaksi_id desc");
```

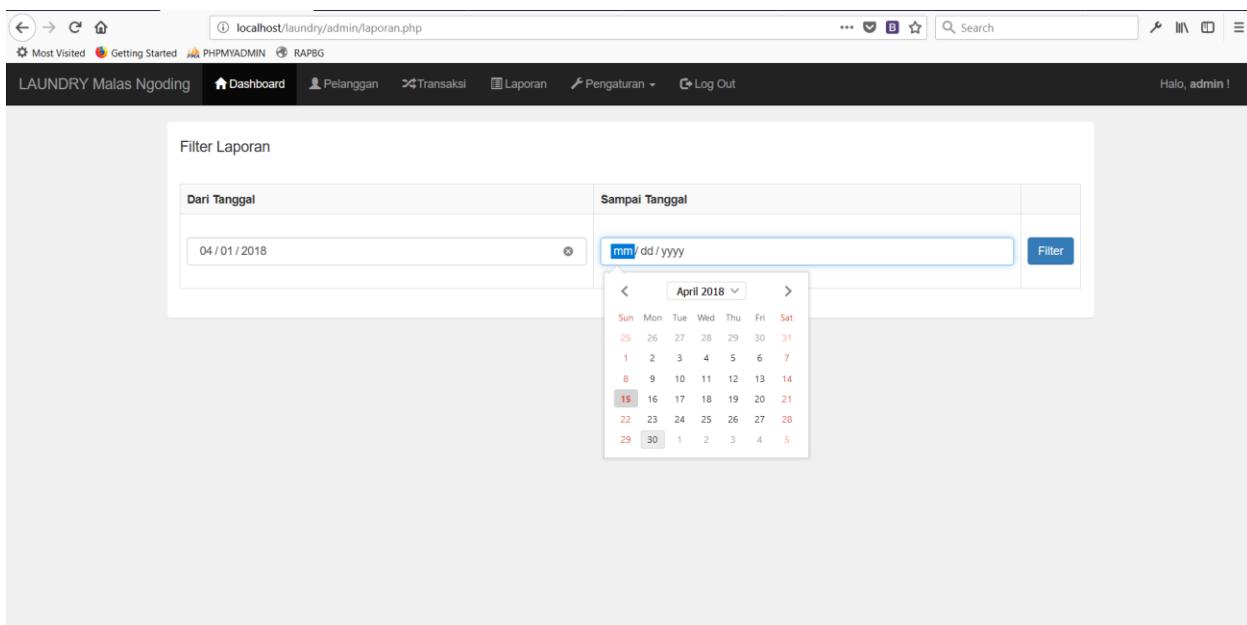
Untuk memberikan perintah tanggal pada query mysql kita bisa menggunakan fungsi date() pada kolumn tabel yang kita inginkan seperti contohnya di atas.

Maksud query di atas adalah kita menentukan transaksi\_tgl yang lebih besar dari data tanggal dari. Dan yang transaksi\_tgl nya lebih kecil dari tanggal sampai.

Kita lihat hasilnya. Coba klik pada menu laporan.



Kemudian isi tanggalnya. Di sini saya mencoba mengisi dari tanggal 1-4-2018. Sampai 30-4-2018. Dan klik filter.



Setelah klik tombol filter, maka akan muncul data transaksi laundry dari tanggal 1 sampai tanggal 30 bulan 4 tahun 2018.

The screenshot shows a web browser window with the URL `localhost/laundry/admin/laporan.php`. Two input fields in the address bar are highlighted with red boxes: `tgl_dari=2018-04-01` and `tgl_sampai=2018-04-30`. A red arrow points from the top right towards these fields. Below the address bar, the page header includes links for Dashboard, Pelanggan, Transaksi, Laporan, Pengaturan, and Log Out, along with a greeting 'Halo, admin !'. The main content area is titled 'Filter Laporan' and contains a form with 'Dari Tanggal' and 'Sampai Tanggal' fields, each with a date input field ('mm / dd / yyyy') and a 'Filter' button. Below this is a section titled 'Data Laporan Laundry dari 2018-04-01 sampai 2018-04-30' containing two buttons: 'CETAK' and 'CETAK PDF'. A table follows, showing transaction details:

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status
1	INVOICE-5	2018-04-15	Sri Dayatun	6	2018-04-21	Rp. 36,000,-	PROSES
2	INVOICE-4	2018-04-15	Sri Dayatun	3	2018-04-18	Rp. 18,000,-	PROSES
3	INVOICE-3	2018-04-15	Sri Dayatun	2	2018-04-17	Rp. 12,000,-	PROSES
4	INVOICE-2	2018-04-15	Sumanto	1	2018-04-16	Rp. 6,000,-	PROSES
5	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI

Cukup mudah bukan. Sebenarnya step by step nya sudah kita pelajari pada part sebelumnya. Di sini kita Cuma memainkan sedikit logika untuk menghasilkan hasil seperti yang kita mau.

## Membuat Print Laporan

Untuk membuat print laporan transaksi laundry ini kita juga telah membuat sebuah tombol yang muncul saat sudah di tentukan tanggal nya seperti contoh di atas.

```
<a target="_blank" href="cetak_print.php?dari=<?php echo $dari;
?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-primary"><i
class="glyphicon glyphicon-print"></i> CETAK</a>
```

Pada tombol cetak di atas kita mengarahkan halaman ke file `cetak_print.php`. sambil mengirim data tanggal yang sudah kita masukkan ke variabel `dari`. Dan juga mengirim data tanggal sampai yang juga sudah kita masukkan dalam variabel `sampai`.

```
href="cetak_print.php?dari=<?php echo $dari; ?>&sampai=<?php echo $sampai;
?>"
```

Dan jangan lupa juga menambahkan

```
target="_blank"
```

Agar halaman cetak laporannya di buka pada tab baru di browser.

Karena kita membuat proses print laporannya dalam file cetak\_print.php seperti di atas. Maka sekarang buat sebuah file baru dengan nama "cetak\_print.php" dalam folder admin.

cetak\_print.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Sistem Informasi Laundry Malas Ngoding -  

    WWW.MALASNGODING.COM</title>

    <link rel="stylesheet" type="text/css"  

    href="../assets/css/bootstrap.css">
        <script type="text/javascript" src="../assets/js/jquery.js"></script>
        <script type="text/javascript"  

        src="../assets/js/bootstrap.js"></script>

</head>
<body>
    <!-- cek apakah sudah login -->
    <?php
        session_start();
        if($_SESSION['status']!='login') {
            header("location:../index.php?pesan=belum_login");
        }
    ?>

    <?php
        // koneksi database
        include '../koneksi.php';
    ?>
    <div class="container">

        <center><h2>LAUNDRY " Malas Ngoding "</h2></center>
        <br/>
        <br/>
        <?php
            if(isset($_GET['dari']) && isset($_GET['sampai'])) {

                $dari = $_GET['dari'];
                $sampai = $_GET['sampai'];
            ?>
                <h4>Data Laporan Laundry dari <b><?php echo $dari; ?></b>
                sampai <b><?php echo $sampai; ?></b></h4>
            
```

```

<table class="table table-bordered table-striped">
    <tr>
        <th width="1%">No</th>
        <th>Invoice</th>
        <th>Tanggal</th>
        <th>Pelanggan</th>
        <th>Berat (Kg)</th>
        <th>Tgl. Selesai</th>
        <th>Harga</th>
        <th>Status</th>
    </tr>

    <?php

        // mengambil data pelanggan dari database
        $data = mysqli_query($koneksi,"select * from
pelanggan,transaksi where transaksi_pelanggan=pelanggan_id and
date(transaksi_tgl) > '$dari' and date(transaksi_tgl) < '$sampai' order by
transaksi_id desc");
        $no = 1;
        // mengubah data ke array dan menampilkannya
dengan perulangan while
        while($d=mysqli_fetch_array($data)) {
            ?>
            <tr>
                <td><?php echo $no++; ?></td>
                <td>INVOICE-<?php echo
$d['transaksi_id']; ?></td>
            ?></td>
                <td><?php echo $d['transaksi_tgl'];
?></td>
                <td><?php echo $d['pelanggan_nama'];
?></td>
                <td><?php echo $d['transaksi_berat'];
?></td>
                <td><?php echo
$d['transaksi_tgl_selesai']; ?></td>
                <td><?php echo "Rp.
".number_format($d['transaksi_harga'])." , -"; ?></td>
                <td>
                    <?php
                        if($d['transaksi_status']=="0") {
                            echo "<div class='label
label-warning'>PROSES</div>";
                        } else
                            echo "<div class='label
label-info'>DICUCI</div>";
                        } else
                            echo "<div class='label
label-success'>SELESAI</div>";
                    }
                ?>
            </td>
        }
    </table>

```

```
        </tr>
        <?php
    }
    ?>
</table>
<?php } ?>

</div>

<script type="text/javascript">
    window.print();
</script>

</body>
</html>
```

Jika teman-teman perhatikan pada syntax cetak\_print.php di atas, intinya sama saja seperti cara menampilkan data transaksi menurut tanggal yang telah kita bahas pada sub judul sebelumnya.

Hanya bedanya di sini kita menambahkan syntax javascript untuk membuat perintah print halaman.

```
<script type="text/javascript">
    window.print();
</script>
```

Sekarang kita lihat hasilnya. Setelah teman-teman menentukan tanggal untuk memfilter laporan transaksi laundrynya. Klik pada tombol “cetak”.

The screenshot shows a web-based laundry management system. At the top, there's a navigation bar with links like 'Dashboard', 'Pelanggan', 'Transaksi', 'Laporan', 'Pengaturan', and 'Log Out'. A user 'Halo, admin!' is logged in. Below the navigation is a 'Filter Laporan' section with 'Dari Tanggal' and 'Sampai Tanggal' input fields, both set to 'mm / dd / yyyy'. A 'Filter' button is to the right. Below this is a table titled 'Data Laporan Laundry dari 2018-04-01 sampai 2018-04-30'. The table has columns: No, Invoice, Tanggal, Pelanggan, Berat (Kg), Tgl. Selesai, Harga, and Status. Five rows of data are listed, with the last row having a status of 'DICUCI'. Two buttons at the top of the table, 'CETAK' and 'CETAK PDF', are highlighted with a red box and arrow.

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status
1	iINVOICE-5	2018-04-15	Sri Dayatun	6	2018-04-21	Rp. 36,000,-	PROSES
2	INVOICE-4	2018-04-15	Sri Dayatun	3	2018-04-18	Rp. 18,000,-	PROSES
3	INVOICE-3	2018-04-15	Sri Dayatun	2	2018-04-17	Rp. 12,000,-	PROSES
4	INVOICE-2	2018-04-15	Sumanto	1	2018-04-16	Rp. 6,000,-	PROSES
5	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI

Maka akan muncul halaman cetak atau print. Beserta dengan data transaksi yang sudah di filter.

The screenshot shows a 'Print' dialog box overlaid on a report preview. The dialog box includes fields for 'Printer' (set to 'Microsoft Print to PDF'), 'Status' (Ready), 'Type' (Microsoft Print To PDF), 'Where' (PORTPROMPT), 'Comment', 'Print range' (radio buttons for 'All', 'Pages from 1 to 1', or 'Selection' - the 'All' option is selected), 'Copies' (Number of copies: 1), and 'OK' and 'Cancel' buttons. To the right of the dialog is a preview of the 'LAUNDRY " Malas Ngoding "' report. The report title is 'Data Laporan Laundry dari 2018-04-01 sampai 2018-04-30'. It contains the same table of laundry transactions as the previous screenshot, with the last row showing 'Status: DICUCI'. A red box and arrow highlight the print dialog box.

Selesai sudah proses print laporan transaksi berdasarkan tanggal. Terakhir akan kita lanjutkan dengan membuat proses cetak laporan transaksi ke dalam file pdf.

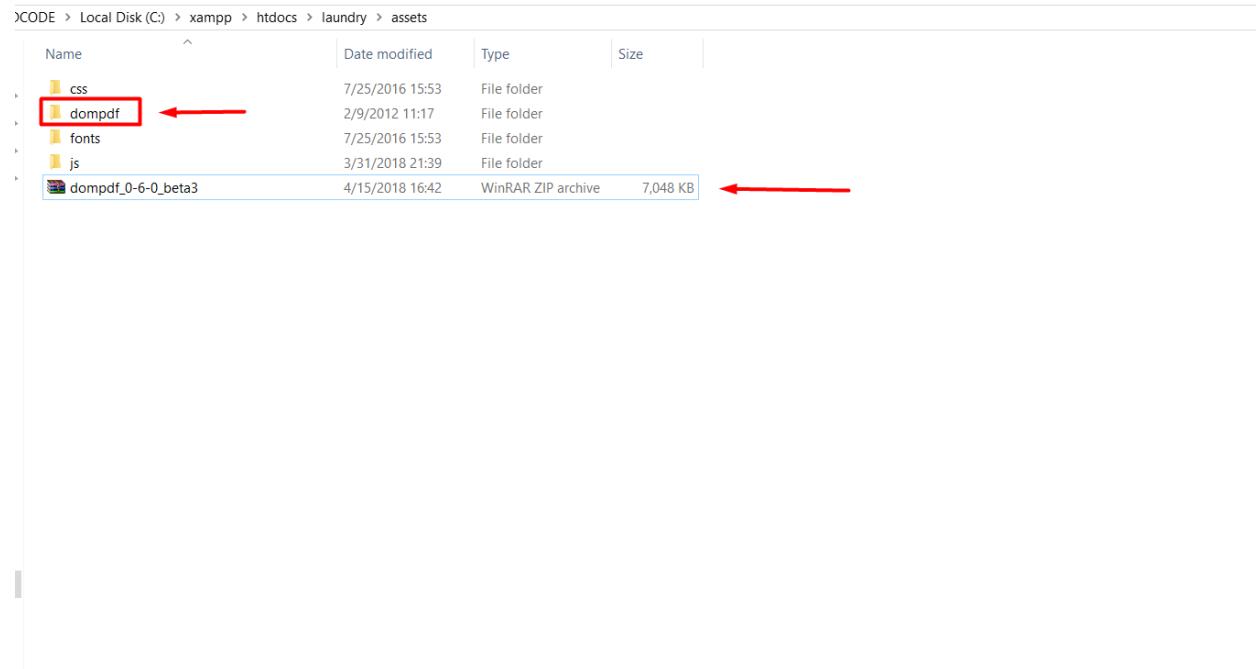
## Membuat Cetak Laporan Ke PDF dengan DOMPDF

Setelah sebelumnya kita selesai membuat cetak atau print laporan transaksi. Pada tahap ini kita akan membuat proses cetak laporan ke dalam file PDF.

Ada banyak cara membuat cetak laporan PDF di PHP. Namun pada panduan ini kita akan menggunakan salah satu nya saja. Yaitu kita akan menggunakan library DOMPDF.

DOMPDF adalah library PHP yang sengaja di buat oleh developer di luar sana untuk memudahkan kita membuat file PDF dengan PHP. Teman-teman bisa mendownload DOMPDF di [https://github.com/downloads/dompdf/dompdf/dompdf\\_0-6-0\\_beta3.zip](https://github.com/downloads/dompdf/dompdf/dompdf_0-6-0_beta3.zip).

Setelah di download, copy kan ke dalam folder assets pada project kita. Kemudian di ekstrak



Perhatikan link pada tombol “cetak pdf”.

```
<a target="_blank" href="cetak_pdf.php?dari=<?php echo $dari; ?>&sampai=<?php echo $sampai; ?>" class="btn btn-sm btn-primary"><i class="glyphicon glyphicon-print"></i> CETAK PDF</a>
```

Pada link tersebut kita mengarahkan ke file cetak\_pdf.php. sambil mengirimkan data tanggal dari dan tanggal sampai. Sama seperti pada saat membuat proses print laporan sebelumnya yang telah kita bahas.

Buat sebuah file baru dengan nama file cetak\_pdf.php dalam folder admin.

### cetak\_pdf.php

```
<?php
// menghubungkan dengan dompdf
require_once("../assets/dompdf/dompdf_config.inc.php");

// koneksi database
include '../koneksi.php';

$html = '<!DOCTYPE html>';
$html .= '<html>';
$html .= '<head>';
$html .= '<title>Sistem Informasi Laundry Malas Ngoding - WWW.MALASNGODING.COM</title>';
$html .= '</head>';
$html .= '<body>';
$html .= '<center><h2>LAUNDRY " Malas Ngoding "</h2></center>';

$dari = $_GET['dari'];
$ Sampai = $_GET['sampai'];

$html .= '<h4>Data Laporan Laundry dari <b>' . $dari . '</b> sampai <b>' . $ Sampai . '</b></h4>';
$html .= '<table border="1" width="100%">';
$html .= '<tr>';
$html .= '<th width="1%">No</th>';
$html .= '<th>Invoice</th>';
$html .= '<th>Tanggal</th>';
$html .= '<th>Pelanggan</th>';
$html .= '<th>Berat (Kg)</th>';
$html .= '<th>Tgl. Selesai</th>';
$html .= '<th>Harga</th>';
$html .= '<th>Status</th>          ';
$html .= '</tr>';

$data = mysqli_query($koneksi, "select * from pelanggan,transaksi where transaksi_pelanggan=pelanggan_id and date(transaksi_tgl) > '$dari' and date(transaksi_tgl) < '$sampai' order by transaksi_id desc");
$no = 1;

while ($d=mysqli_fetch_array($data)) {
    $html .= '<tr>';
    $html .= '<td>' . $no++ . '</td>';
    $html .= '<td>INVOICE-' . $d['transaksi_id'] . '</td>';
    $html .= '<td>' . $d['transaksi_tgl'] . '</td>';
```

```

$html .= '<td>'.$d['pelanggan_nama'].'</td>';
$html .= '<td>'.$d['transaksi_berat'].'</td>';
$html .= '<td>'.$d['transaksi_tgl_selesai'].'</td>';
$html .= '<td> Rp. '.number_format($d["transaksi_harga"]).' ,-</td>';
$html .= '<td>';

if($d['transaksi_status']=="0") {
    $html .= "PROSES";
} else if($d['transaksi_status']=="1") {
    $html .= "DICUCI";
} else if($d['transaksi_status']=="2") {
    $html .= "SELESAI";
}

$html .= '</td>';
$html .= '</tr>';

}

$html .= '</table>';
$html .= '</body>';
$html .= '</html>';

$dompdf = new DOMPDF();
$dompdf->set_paper('a4','landscape');
$dompdf->load_html($html);
$dompdf->render();
$dompdf->stream('laporan dari'.$dari.' sampai '.$sampai.'.pdf');?>

```

Pertama kita hubungkan dulu dengan file dompdf\_config.php yang terletak dalam folder dompdf yang telah kita download.

```
require_once("../assets/dompdf/dompdf_config.inc.php");
```

Proses selanjutnya sama saja dengan proses menampilkan data saat membuat proses print laporan. Bedanya hanya di sini kita mengimpen semua yang ingin kita tampilan dalam folder “\$html”.

Penggunaan tanda “. (titik)” pada variabel html di atas adalah untuk menggabungkan atau menambah isi dari variabel tersebut. Seperti contoh yang telah kita buat pada file cetak\_pdf fi atas.

Dan terakhir pada syntax berikut adalah perintah dari dompdf yang bisa kita gunakan untuk membuat file menjadi pdf. Kita bisa menentukan ukuran kertas dan posisi kertas. Menentukan nama laporan saat menjadi pdf.

```

$dompdf = new DOMPDF();
$dompdf->set_paper('a4','landscape');
$dompdf->load_html($html);
$dompdf->render();
$dompdf->stream('laporan dari'.$dari.' sampai '.$sampai.'.pdf');

```

Semua data yang sudah dimasukkan ke dalam variabel html tadi kita load ke dompdf dengan function load\_html() seperti pada syntax di atas. Di sini file laporan pdf yang kita buat akan bernama " laporan\_dari\_(tanggal dari)\_sampai\_(tanggal sampai).pdf ".

Mari kita lihat hasilnya.

Silahkan klik pada tombol cetak pdf.

localhost/laundry/admin/laporan.php?tgl\_dari=2018-04-01&tgl\_sampai=2018-04-30

LAUNDRY Malas Ngoding

Filter Laporan

Dari Tanggal	Sampai Tanggal	Filter
mm / dd / yyyy	mm / dd / yyyy	

Data Laporan Laundry dari 2018-04-01 sampai 2018-04-30

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status
1	INVOICE-5	2018-04-15	Sri Dayatun	6	2018-04-21	Rp. 36,000,-	PROSES
2	INVOICE-4	2018-04-15	Sri Dayatun	3	2018-04-18	Rp. 18,000,-	PROSES
3	INVOICE-3	2018-04-15	Sri Dayatun	2	2018-04-17	Rp. 12,000,-	PROSES
4	INVOICE-2	2018-04-15	Sumanto	1	2018-04-16	Rp. 6,000,-	PROSES
5	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI

maka hasilnya

file:///C:/Users/asus/Downloads/Documents/laporan\_dari2018-04-01\_sampai\_2018-04-30.pdf

**LAUNDRY " Malas Ngoding "**

Data Laporan Laundry dari 2018-04-01 sampai 2018-04-30

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status
1	INVOICE-5	2018-04-15	Sri Dayatun	6	2018-04-21	Rp. 36,000,-	PROSES
2	INVOICE-4	2018-04-15	Sri Dayatun	3	2018-04-18	Rp. 18,000,-	PROSES
3	INVOICE-3	2018-04-15	Sri Dayatun	2	2018-04-17	Rp. 12,000,-	PROSES
4	INVOICE-2	2018-04-15	Sumanto	1	2018-04-16	Rp. 6,000,-	PROSES
5	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI

Nah, proses pembuatan cetak laporan ke dalam bentuk PDF pun selesai.

## Rekapitulasi Data Pada Dashboard Admin

Setelah semua proses selesai kita buat, sekarang kita akan memperbaiki halaman dashboard admin. Dimana sebelumnya halaman dashboard (index.php) masih kosong, supaya lebih keren, kita akan menampilkan data rekap transaksi pada halaman dashboard.

Berupa riwayat data transaksi terakhir, kita juga menampilkan jumlah pelanggan, jumlah cucian berdasarkan statusnya. Yaitu cucian yang berstatus sedang di proses atau sedang di cuci, cucian yang berstatus sudah siap di cuci dan sudah bisa di ambil oleh pelanggan. Dan juga menampilkan jumlah cucian yang sudah di ambil (transaksi selesai).

Buka kembali file index.php yang sudah kita buat sebelumnya.

### Index.php

```
<?php include 'header.php'; ?>

<?php
// koneksi database
include '../koneksi.php';
?>

<div class="container">
    <div class="alert alert-info text-center">
        <h4 style="margin-bottom: 0px"><b>Selamat datang!</b> di sistem
informasi laundry malas ngoding.</h4>
    </div>

    <div class="panel">
        <div class="panel-heading">
            <h4>Dashboard</h4>
        </div>
        <div class="panel-body">

            <div class="row">
                <div class="col-md-3">
                    <div class="panel panel-primary">
                        <div class="panel-heading">
                            <h1>
                                <i class="glyphicon glyphicon-user"></i>
                            <span class="pull-right">
                                <?php
                                    $pelanggan =
                                    echo
                                    mysqli_query($koneksi, "select * from pelanggan");
                                </?php>
                            </span>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

mysqli_num_rows($pelanggan);
                                ?>
                            </span>
                        </h1>
                        Jumlah Pelanggan
                    </div>
                </div>
            </div>

<div class="col-md-3">
    <div class="panel panel-warning">
        <div class="panel-heading">
            <h1>
                <i class="glyphicon glyphicon-retweet"></i>
                <span class="pull-right">
                    <?php
                        $proses =
                    mysqli_query($koneksi,"select * from transaksi where transaksi_status='0'");
                    echo
                    mysqli_num_rows($proses);
                    ?>
                </span>
            </h1>
            Jumlah Cucian Di Proses
        </div>
    </div>
</div>

<div class="col-md-3">
    <div class="panel panel-info">
        <div class="panel-heading">
            <h1>
                <i class="glyphicon glyphicon-info-sign"></i>
                <span class="pull-right">
                    <?php
                        $proses =
                    mysqli_query($koneksi,"select * from transaksi where transaksi_status='1'");
                    echo
                    mysqli_num_rows($proses);
                    ?>
                </span>
            </h1>
            Jumlah Cucian Siap Ambil
        </div>
    </div>
</div>

<div class="col-md-3">
    <div class="panel panel-success">
        <div class="panel-heading">
            <h1>
                <i class="glyphicon glyphicon-ok-circle"></i>

```

```

        <span class="pull-right">

            <?php
            $proses =
mysqli_query($koneksi,"select * from transaksi where transaksi_status='2'");
echo
mysqli_num_rows($proses);
                ?>
            </span>
        </h1>
        Jumlah Cucian Selesai
    </div>
</div>
</div>

</div>
</div>

<div class="panel">
    <div class="panel-heading">
        <h4>Riwayat Transaksi Terakhir</h4>
    </div>
    <div class="panel-body">
        <table class="table table-bordered table-striped">
            <tr>
                <th width="1%">No</th>
                <th>Invoice</th>
                <th>Tanggal</th>
                <th>Pelanggan</th>
                <th>Berat (Kg)</th>
                <th>Tgl. Selesai</th>
                <th>Harga</th>
                <th>Status</th>
            </tr>

            <?php
            // mengambil data pelanggan dari database
            $data = mysqli_query($koneksi,"select * from
pelanggan,transaksi where transaksi_pelanggan=pelanggan_id order by
transaksi_id desc limit 7");
            $no = 1;
            // mengubah data ke array dan menampilkannya dengan
perulangan while
            while($d=mysqli_fetch_array($data)){
                ?>
                <tr>
                    <td><?php echo $no++; ?></td>
                    <td>INVOICE-<?php echo
$d['transaksi_id']; ?></td>
                ?></td>
                    <td><?php echo $d['transaksi_tgl'];
?></td>
                ?></td>
                    <td><?php echo $d['pelanggan_nama'];
?></td>
                ?></td>
                    <td><?php echo $d['transaksi_berat'];
?></td>
                ?></td>
            }
        </table>
    </div>
</div>

```

```

        <td><?php echo
$d['transaksi_tgl_selesai']; ?></td>
        <td><?php echo "Rp.
".number_format($d['transaksi_harga']) . "-" ; ?></td>
        <td>
            <?php
            if($d['transaksi_status']=="0") {
                echo "<div class='label
label-warning'>PROSES</div>";
            } else
            if($d['transaksi_status']=="1") {
                echo "<div class='label
label-info'>DICUCI</div>";
            } else
            if($d['transaksi_status']=="2") {
                echo "<div class='label
label-success'>SELESAI</div>";
            }
        ?>

        </td>
    </tr>
    <?php
}
?>
</table>
</div>
</div>

</div>

<?php include 'footer.php'; ?>

```

Perhatikan pada syntax di atas. Di sana kita menampilkan jumlah pelanggan dengan query dan syntax berikut.

```

<div class="row">
    <div class="col-md-3">
        <div class="panel panel-primary">
            <div class="panel-heading">
                <h1>
                    <i class="glyphicon glyphicon-user"></i>
                    <span class="pull-right">
                        <?php
                        $pelanggan =
mysql_query($koneksi,"select * from pelanggan");
                        echo mysqli_num_rows($pelanggan);
                    ?>
                </span>
            </h1>
            Jumlah Pelanggan
        </div>
    </div>

```

```

        </div>
    </div>
    <div class="col-md-3">
        <div class="panel panel-warning">
            <div class="panel-heading">
                <h1>
                    <i class="glyphicon glyphicon-retweet"></i>
                    <span class="pull-right">

                        <?php
                            $proses = mysqli_query($koneksi,"select *
from transaksi where transaksi status='0'" );
                            echo mysqli_num_rows($proses);
                            ?>
                    </span>
                </h1>
                Jumlah Cucian Di Proses
            </div>
        </div>
    </div>

<div class="col-md-3">
    <div class="panel panel-info">
        <div class="panel-heading">
            <h1>
                <i class="glyphicon glyphicon-info-sign"></i>
                <span class="pull-right">

                    <?php
                        $proses = mysqli_query($koneksi,"select *
from transaksi where transaksi_status='1'" );
                        echo mysqli_num_rows($proses);
                        ?>
                    </span>
                </h1>
                Jumlah Cucian Siap Ambil
            </div>
        </div>
    </div>

<div class="col-md-3">
    <div class="panel panel-success">
        <div class="panel-heading">
            <h1>
                <i class="glyphicon glyphicon-ok-circle"></i>
                <span class="pull-right">

                    <?php
                        $proses = mysqli_query($koneksi,"select *
from transaksi where transaksi_status='2'" );
                        echo mysqli_num_rows($proses);
                        ?>
                </span>
            </h1>
            Jumlah Cucian Selesai
        </div>
    </div>

```

```
</div>  
</div>
```

Contohnya untuk menghitung jumlah pelanggan, teman-teman bisa menggunakan query seperti berikut.

```
<?php  
$pelanggan = mysqli_query($koneksi,"select * from pelanggan");  
echo mysqli_num_rows($pelanggan);  
?>
```

Pertama kita dapatkan dulu data pelanggannya dengan menggunakan query seperti biasa, yaitu "Select \* from pelanggan". Kemudian untuk menghitung jumlahnya kita bisa menggunakan fungsi mysqli\_num\_rows().

Pada tabel bagian bawah kita menampilkan 7 data transaksi terbaru dengan menggunakan query berikut.

```
$data = mysqli_query($koneksi,"select * from pelanggan,transaksi where  
transaksi_pelanggan=pelanggan_id order by transaksi_id desc limit 7");
```

Oke langsung saja kita lihat hasilnya. Klik pada menu dashboard.

The screenshot shows a web browser displaying a dashboard for a laundry management system. The URL in the address bar is `localhost/laundry/admin/index.php`. The dashboard has a dark header with the text "LAUNDRY Malas Ngoding" and "Halo, admin !". Below the header, there's a message "Selamat datang! di sistem informasi laundry malas ngoding." A red box highlights the top section showing four summary statistics:

Jumlah Pelanggan	2
Jumlah Cucian Di Proses	4
Jumlah Cucian Siap Ambil	1
Jumlah Cucian Selesai	0

Below this, a red arrow points to a table titled "Riwayat Transaksi Terakhir" (Recent Transaction History) containing the following data:

No	Invoice	Tanggal	Pelanggan	Berat (Kg)	Tgl. Selesai	Harga	Status
1	INVOICE-5	2018-04-15	Sri Dayatun	6	2018-04-21	Rp. 36,000,-	PROSES
2	INVOICE-4	2018-04-15	Sri Dayatun	3	2018-04-18	Rp. 18,000,-	PROSES
3	INVOICE-3	2018-04-15	Sri Dayatun	2	2018-04-17	Rp. 12,000,-	PROSES
4	INVOICE-2	2018-04-15	Sumanto	1	2018-04-16	Rp. 6,000,-	PROSES
5	INVOICE-1	2018-04-06	Sumanto	3	2018-04-08	Rp. 18,000,-	DICUCI

Nah, selesai. Jumlah pelanggan, jumlah transaksi yang sedang di proses, jumlah transaksi yang siap, dan jumlah transaksi yang selesai pun berhasil kita tampilkan. Sekaligus menampilkan riwayat transaksi terakhir atau terbaru pada bagian bawahnya.

**SELESAI**

# TUTORIAL REKOMENDASI SELANJUTNYA

Setelah teman-teman berhasil menguasai materi tutorial “ **BELAJAR PHP & MYSQLI DARI DASAR SAMPAI MEMBUAT SISTEM INFORMASI LAUNDRY** ” ini, kami merekomendasikan sebuah tutorial yang bisa dipelajari untuk materi selanjutnya. Agar kemampuan ngoding teman-teman tidak *stack* di materi dasar.

Teman-teman bisa meningkatkan kemampuan ngoding teman-teman dengan mempelajari Framework PHP. Apa itu framework? Dan bagaimana cara kerjanya? Teman-teman bisa mempelajarinya pada tutorial “ **BELAJAR CEPAT CODEIGNITER DENGAN STUDI KASUS MEMBUAT APLIKASI RENTAL MOBIL** ” berikut. Tidak hanya mendapatkan materinya saja, teman-teman juga akan mendapat lengkap *source code aplikasi rental mobil* hasil jadinya.



Berikut adalah list materi dari Ebook “**BELAJAR CEPAT CODEIGNITER DENGAN STUDI KASUS MEMBUAT APLIKASI RENTAL MOBIL**” .

### Tutorial Codeigniter dasar

- Dasar Bagian 1 : Berkenalan Dengan Framework Codeigniter
- Dasar Bagian 2 : Cara Instalasi Framework Codeigniter
- Dasar Bagian 3 : Mengenal Uri Segment Pada Codeigniter
- Dasar Bagian 4 : Pengenalan Controller Codeigniter
- Dasar Bagian 5 : Pengenalan View Pada Codeigniter
- Dasar Bagian 6 : Membuat Template Web Sederhana (Templating) Pada Codeigniter
- Dasar Bagian 7 : Form Validation Pada Codeigniter
- Dasar Bagian 8 : Upload File Dengan Codeigniter
- Dasar Bagian 9 : Membuat Download File Dengan Codeigniter
- Dasar Bagian 10 : Membuat Library Codeigniter Sendiri
- Dasar Bagian 11 : Menghilangkan Index.php Pada Codeigniter
- Dasar Bagian 12 : Mengenal Model, Koneksi Database Codeigniter dan MySQL
- Dasar Bagian 13 : Membuat CRUD Dengan Codeigniter
- Dasar Bagian 14 : Membuat Pagination Dengan Codeigniter
- Dasar Bagian 15 : Membuat Login Dengan Codeigniter

### Tutorial Codeigniter Lanjutan Studi Kasus Membuat Aplikasi Rental Mobil

- Instalasi Codeigniter 3
- Persiapan Membuat Aplikasi Rental Mobil
- Setting Base URL
- Load Library, Helper, dan segala yang dibutuhkan
- Merancang & Membuat Database Aplikasi Rental Mobil dan Konfigurasi database di Codeigniter
- Konfigurasi database pada codeigniter
- Membuat Model Yang Terstruktur dan Efisien Untuk CRUD
- Menghilangkan index.php dari URL Codeigniter
- Integrasi Codeigniter dan Bootstrap
- Membuat login dengan codeigniter, md5 & Pesan Notifikasi
- Menyiapkan Halaman Dashboard Admin
- Membuat Logout

- Membuat Fitur Ganti Password Admin
- Membuat CRUD Data Mobil
- Membuat CRUD Data Kostumer
- Membuat Proses Transaksi Rental Mobil
- Menampilkan Data Transaksi Rental
- Menginput Data Transaksi Rental Mobil
- Membuat Batalkan Transaksi
- Membuat Transaksi Selesai (Konfirmasi)
- Membuat Filter dan Cetak Laporan Rental Mobil
- Menampilkan Data Transaksi Berdasarkan Filter Tanggal
- Membuat Print Laporan
- Membuat Cetak Laporan Ke PDF Menggunakan DOMPDF
- Selesai

#### **Yang di dapatkan :**

- Materi
- Software Pendukung
- Library dan lain-lain (perlengkapan)
- Source Code Aplikasi Rental Mobil
- Konsultasi dengan admin via Whatsapp

Untuk selengkapnya tentang ebook “**BELAJAR CEPAT CODEIGNITER DENGAN STUDI KASUS MEMBUAT APLIKASI RENTAL MOBIL**”, teman-teman bisa langsung membaca deskripsinya di [www.malasngoding.com/shop](http://www.malasngoding.com/shop), atau silahkan kontak admin melalui nomor whatsapp yang tertera di [www.malasngoding.com](http://www.malasngoding.com).