

Menyelam & Menaklukan

Samudra PHP

Oleh : Loka Dwiartara

Suported by :

Direktori Weblogger Indonesia
Blogger Based Indonesia Web Blog Directory
<http://www.direktoriweblog.com>

“Ebook ini didedikasikan untuk generasi **manusia-manusia muda**
yang akan membangunkan tidurnya sang raksasa adidaya. Al-Khilafah System.”

Oleh : **Loka Dwiartara**

Tentang Penulis



Loka Dwiartara, merupakan seorang Praktisi IT yang memiliki spesialisasi di bidang web development ini telah lama bergelut dalam dunia **PHP dan MySQL** semenjak masih kuliah. Saat ini bekerja sebagai programmer *Sco International*, dan bekerja sama dengan **Green Technomedia** sebagai kepala project pembuatan *UBB Online* dan *Smart School System*, juga bekerja sebagai Trainer di **CBS (Cyber Business School - Bogor)** , kampus STT Telematika dan **SMK Swasta** di Bogor. Penulis yang sekaligus merupakan chief e-zine ilmuwebsite ini selain gemar melakukan sharing khususnya di bidang web development dan open source, juga senang sekali bermain-main dengan **web bot technology**, membuat satu aplikasi otomatis yang berfungsi untuk mencari informasi sekaligus menyebarkannya. Untuk info lebih lanjut mengenai penulis anda dapat mengunjungi websitenya langsung <http://www.ilmuwebsite.com> maupun mengirim email alkemail@gmail.com

Pengantar

Banyak orang mencari referensi mengenai bagaimana membuat website / membuat aplikasi berbasis website menggunakan PHP dan MySQL, sehingga banyak orang pula tak tanggung-tanggung membeli banyak buku untuk memberikan informasi-informasi hingga seseorang itu menjadi mahir dalam ber-PHP ria. Namun jika ada yang gratis mengapa harus membelinya? Ada yang berpendapat, ya untuk referensi saja, semakin banyak buku semakin cepat membuat isi di kepala meledak.

Ebook PHP ini membahas dasar-dasar bagaimana menggunakan PHP dan MySQL dalam membuat website yang dinamis maupun bagaimana membuat aplikasi berbasis website. Dengan basic yang kuat di harapkan para pembaca nanti menjadi tercerahkan akibat membaca ebook ini.

Penulis sadar banyak sekali ketidaksempurnaan yang muncul dari ebook PHP ini, oleh karena itu mohon kritik dan sarannya dilayangkan ke email penulis. Sekian dari penulis. Selamat Menikmati Ebook PHP “Menyelam dan Menaklukan Samudra PHP”

Content

Bab 1	: Perkenalan Instalasi dan Konfigurasi	1
Bab 2	: Bekerja dengan PHP	16
Bab 3	: Menggunakan PHP dan MySQL.....	69
Bab 4	: Menampilkan Data Menggunakan Table	91
Bab 5	: Bekerja dengan Form.....	95
Bab 6	: Mengedit Database dengan Form.....	117
Bab 7	: Lebih Jauh dengan Form.....	136
Bab 8	: Mengirim Email Menggunakan PHP.....	148
Bab 9	: PHP Authentication.....	152
Bab 10	: Studi Kasus Membuat Sistem Informasi.....	156

Bab 1:

Perkenalan, Installasi dan Konfigurasi

Perkenalan

1.1 Sebelum Belajar PHP

Sebelum lebih lanjut mempelajari bagaimana membuat web dinamis, atau aplikasi berbasis web menggunakan PHP dan MySQL, penulis menyarankan anda untuk terlebih dahulu mengerti bagaimana membangun web menggunakan HTML. Ini digunakan sebagai bekal awal agar nantinya anda tidak kesulitan dalam mengenali tag-tag HTML yang digunakan dalam membuat website dinamis di ebook PHP ini. Perlu diketahui PHP yang akan dipelajari dalam ebook PHP ini adalah PHP versi ke 5. Karena pada dasarnya tidak terlalu jauh berbeda antara versi 5 dengan PHP versi terbaru yakni PHP 6, perbedaan yang mendasar hanyalah fitur OOP-nya saja, dan OOP tidak dibahas secara detil dalam ebook PHP ini. Namun mengingat cakupan PHP begitu luas, saya akan menghadirkannya nanti di ebook free edisi selanjutnya.

1.2 Perkenalan PHP, MySQL, Apache, dan OpenSource

Sebelum mengenal PHP, MySQL, Apache, perlu diketahui ketiga aplikasi yang disebutkan tadi merupakan aplikasi open source. Aplikasi OpenSource merupakan aplikasi berlisensi GPL (General Public License) yang diperuntukkan secara bebas digunakan oleh Masyarakat Internasional dan sifatnya gratis, open source berarti kode

sumbernya terbuka, dimana seorang pengguna, maupun pengembang, dapat mengostumisasi, dan mengembangkan kode tersebut secara bebas.

Opensource memungkinkan para programmer dunia membuat aplikasi tersebut mengalami perkembangan yang sangat pesat sebagaimana halnya Linux, linux mengalami perkembangan yang begitu cepat, dan pesat jauh berbeda dengan metode yang diterapkan oleh Microsoft yakni Closed Source pada aplikasi-aplikasi buatannya.

1.2.1 Apa itu PHP

PHP Pertama kali ditemukan pada 1995 oleh seorang Software Developer bernama Rasmus Lerdorf. Ide awal PHP adalah ketika itu Radmus ingin mengetahui jumlah pengunjung yang membaca resume onlinenya. script yang dikembangkan baru dapat melakukan dua pekerjaan, yakni merekam informasi visitor, dan menampilkan jumlah pengunjung dari suatu website. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia web saat ini. Kemudian, dari situ banyak orang di milis mendiskusikan script buatan Rasmus Lerdorf, hingga akhirnya rasmus mulai membuat sebuah tool/script, bernama Personal Home Page (PHP).

Kebutuhan PHP sebagai tool yang serba guna membuat Lerdorf melanjutkan untuk mengembangkan PHP hingga menjadi suatu bahasa tersendiri yang mungkin dapat mengkonversikan data yang di inputkan melalui Form HTML menjadi suatu variable, yang dapat dimanfaatkan oleh sistem lainnya. Untuk merealisasikannya, akhirnya Lerdorf mencoba mengembangkan PHP menggunakan bahasa C ketimbang menggunakan Perl. Tahun 1997, PHP versi 2.0 di rilis, dengan nama Personal Home Page Form Interpreter (PHP-FI). PHP Semakin popular, dan semakin diminati oleh programmer web dunia.

Rasmus Lerdorf benar-benar menjadikan PHP sangat populer, dan banyak sekali Team Developer yang ikut bergabung dengan Lerdorf untuk mengembangkan PHP hingga menjadi seperti sekarang, Hingga akhirnya dirilis versi ke 3-nya, pada Juni 1998, dan tercatat lebih dari 50.000 programmer menggunakan PHP dalam membuat website dinamis.

Pengembangan demi pengembangan terus berlanjut, ratusan fungsi ditambahkan sebagai fitur dari bahasa PHP, dan di awal tahun 1999, netcraft mencatat, ditemukan 1.000.000 situs di dunia telah menggunakan PHP. Ini membuktikan bahwa PHP merupakan bahasa yang paling populer digunakan oleh dunia web development. Hal ini mengagetkan para developernya termasuk Rasmus sendiri, dan tentunya sangat diluar dugaan sang pembuatnya. Kemudian Zeev Suraski dan Andi Gutsman selaku core developer (programmer inti) mencoba untuk menulis ulang PHP Parser, dan diintegrasikan dengan menggunakan Zend scripting engine, dan mengubah jalan alur operasi PHP. Dan semua fitur baru tersebut di rilis dalam PHP 4.

13 Juli 2004, evolusi PHP, PHP telah mengalami banyak sekali perbaikan disegala sisi, dan wajar jika netcraft mengumumkan PHP sebagai bahasa web populer didunia, karena tercatat 19 juta domain telah menggunakan PHP sebagai server side scriptingnya. PHP saat ini telah Mendukung XML dan Web Services, Mendukung SQLite. Tercatat lebih dari 19 juta domain telah menggunakan PHP sebagai server scriptingnya. Benar-benar PHP sangat mengejutkan.

Yang menjadikan PHP berbeda dengan HTML adalah proses dari PHP itu sendiri. HTML merupakan bahasa statis yang apabila kita ingin merubah konten/isinya maka yang harus dilakukan pertama kali nya adalah, membuka file-nya terlebih dahulu, kemudian menambahkan isi kedalam file tersebut. Beda hal nya dengan PHP. Bagi anda yang pernah menggunakan CMS seperti wordpress atau joomla yang dibangun dengan PHP tentunya, ketika akan menambahkan konten kedalam website, anda tinggal masuk kedalam halaman admin, kemudian pilih new artikel untuk membuat halaman/content baru. Artinya hal ini, seorang user tidak berhubungan langsung dengan scriptnya. Sehingga seorang pemula sekalipun dapat menggunakan aplikasi seperti itu.

Keunggulan PHP :

1. Gratis

Apa yang membuat PHP begitu berkembang sangat pesat?? hingga jutaan domain menggunakan PHP, begitu populernya PHP?? Jawabannya adalah karena PHP itu gratis. Saya sendiri menyukai bahasa yang satu ini selain mudah juga karena gratis.

2. Cross platform

Artinya dapat di gunakan di berbagai sistem operasi, mulai dari linux, windows, mac os dan os yang lain.

3. Mendukung banyak database

PHP telah mendukung banyak database, ini mengapa banyak developer web menggunakan PHP Adabas D Adabas D, dBase dBase, Empress Empress, FilePro (read-only) FilePro (read-only) Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase mSQL, Direct MS-SQL, MySQL MySQL, ODBC, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL SQLite, Solid, Sybase, Velocis, Unix dbm.

4. On The Fly

PHP sudah mendukung on the fly, artinya dengan php anda dapat membuat document

text, Word, Excel, PDF, menciptakan image dan flash, juga menciptakan file-file seperti zip, XML, dan banyak lagi.

1.2.2 Apa itu MySQL

MySQL adalah Database

Database sendiri merupakan suatu jalan untuk dapat menyimpan berbagai informasi dengan membaginya berdasarkan kategori-kategori tertentu. Dimana informasi-informasi tersebut saling berkaitan, satu dengan yang lainnya.

MySQL bersifat RDBMS (Relational Database Management System)

RDBMS memungkinkan seorang admin dapat menyimpan banyak informasi ke dalam table-table, dimana table-table tersebut saling berkaitan satu sama lain. Keuntungan RDBMS sendiri adalah kita dapat memecah database kedalam table-table yang berbeda. setiap table memiliki informasi yang berkaitan dengan table yang lainnya.

Mengapa Menggunakan MySQL

1. Gratis

Sama dengan PHP, MySQL bersifat opensource, semua orang bebas menggunakannya tanpa harus membayar sepeser pun

2. Cross Platform

MySQL dapat digunakan under windows, ataupun under linux.

3. Lengkap dan Cepat

Pasangan yang cocok dengan PHP. Wajar jika banyak hosting saat ini mendukung adanya PHP dan MySQL karena kecepatan, gratis, dan dapat di jalankan di sistem operasi manapun.

1.2.3 Apa itu Apache

Mengapa menggunakan Apache?

1. Merupakan webserver.

Tempat php engine/processor berada. Tempat meletakkan file-file php dan database. Ketika user melakukan request http:// membuka suatu halaman, disinilah apache bekerja. Menjawab request tersebut dengan menampilkan halaman yang diminta.

2. Apache sama seperti PHP dan MySQL, Gratis.

3. Cross Platform

Perbedaan fungsi antara PHP, MySQL dan Apache adalah, PHP merupakan bahasanya, MySQL adalah databasenya, dan Apache merupakan webserver yang dapat mengeksekusi script php dan menampilkannya kepada user, dan melalui apache lah php dapat mengolah data dan menyimpan data tersebut ke dalam database.

1.3 Mengapa PHP? dan Bagaimana Masa Depan PHP?

PHP telah benar-benar mempengaruhi dunia internet saat ini. Facebook yang anda kenal selama ini dibangun menggunakan PHP, begitu juga dengan Yahoo, dan masih banyak lagi yang website populer lain yang menggunakan php sebagai bahasa intinya. Programmer sistem informasi saat ini lebih cenderung membangun sistem informasi berbasis website, ketimbang desktop application, karena fitur yang dihasilkan sama persis, installasi mudah, dapat dijalankan dibanyak PC, dengan spek yang minim.

Saya ketika masih menjadi mahasiswa pun pernah mendapatkan tawaran bekerja sebagai developer sistem informasi kepegawaian pada salah satu media cetak yang cukup besar, dari pihak mereka sendiri pun mengutamakan PHP ketimbang bahasa/tools programming yang lainnya.

“Masa Depan PHP? PHP sekarang sudah menguasai dunia dengan semua kemudahan dan kegratisannya.”

Installasi

1.4 Installasi PHP, MySQL, & Apache dengan xampp

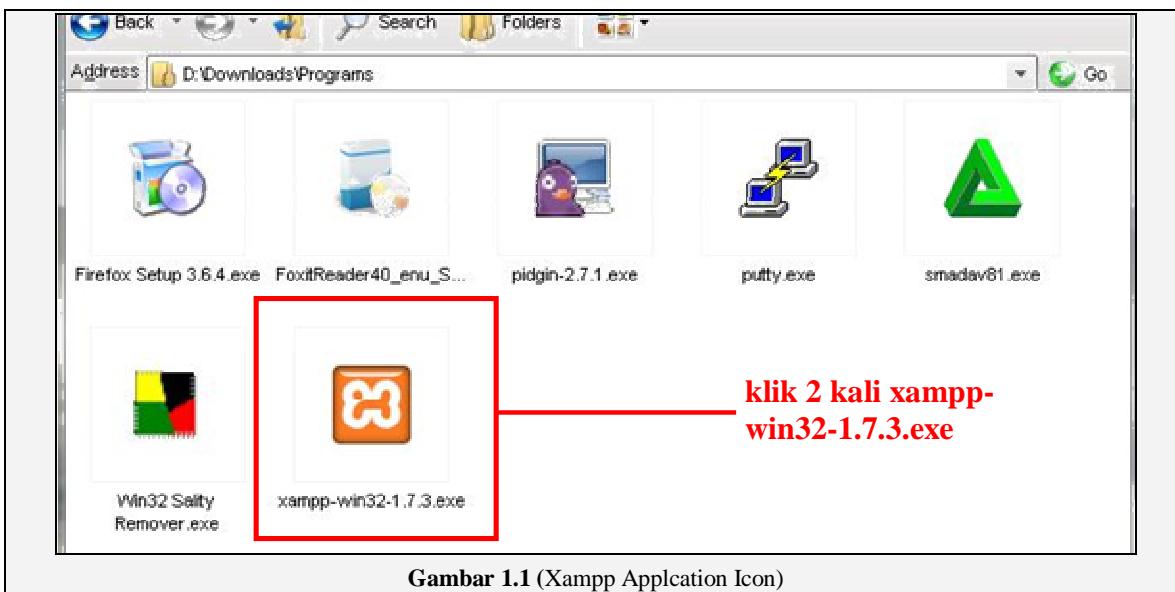
Pada bagian ini akan dijelaskan bagaimana anda dapat menjalankan PHP, MySQL dan Apache secara bersamaan.

Ada 2 opsi sebetulnya, pertama anda menginstall ketiga aplikasi tersebut satu persatu, dan kemudian mengkonfigurasikan satu persatu aplikasi-aplikasi tersebut sampai semuanya terintegrasi, tentunya hal ini benar-benar merepotkan, dan opsi yang kedua adalah, saat ini telah banyak web server paketan yang didalamnya telah berisi PHP, MySQL, dan Apache, dan yang lebih mengasyikan lagi, paket ini gratis, beberapa paket yang terkenal adalah, Sokkit, wammp, xampp, easypHP, apachetriad, dan masih banyak lagi, namun saya sendiri lebih prefer menggunakan xampp, karena sekali install, semua beres, tanpa harus konfigurasi lagi, yang menyenangkan adalah xampp sudah menyediakan Zend PHP Framework dan PHP Pear.

Oke kita langsung saja kepada praktik dalam menginstall xampp, sebuah paket yang berisi PHP, MySQL, Apache secara lengkap. xampp sendiri bisa langsung anda download di <http://www.apachefriends.com>. Anda tinggal mengekstraknya lalu klik dua kali.

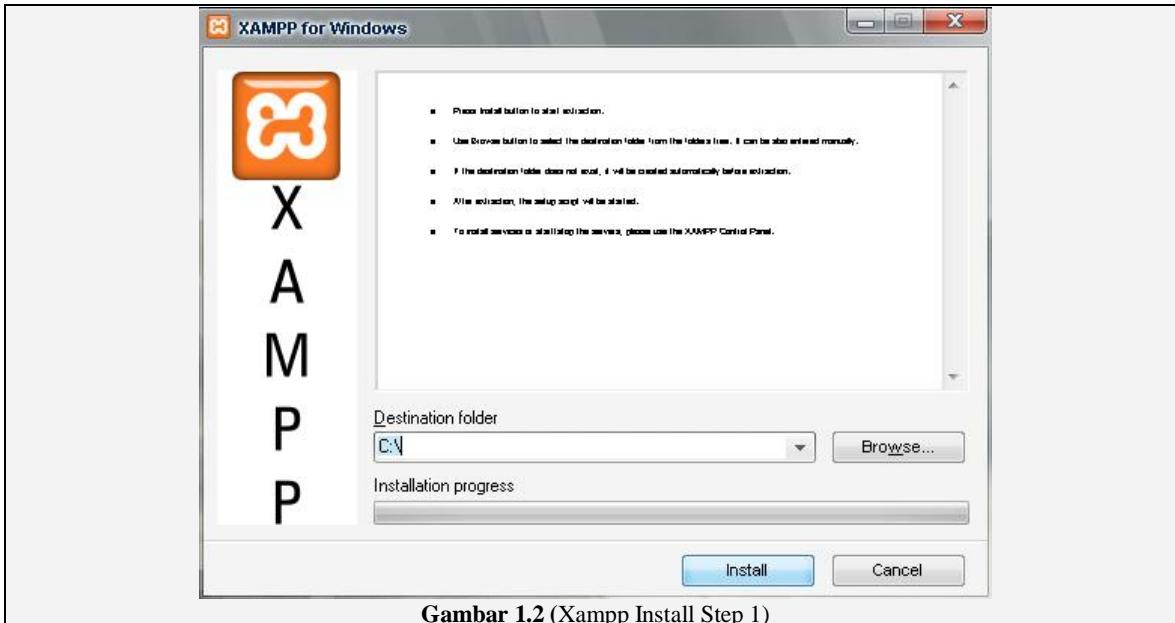
Langkah 1:

Double klik xampp yang sudah anda download, yakni xampp-win32-1.7.3.exe



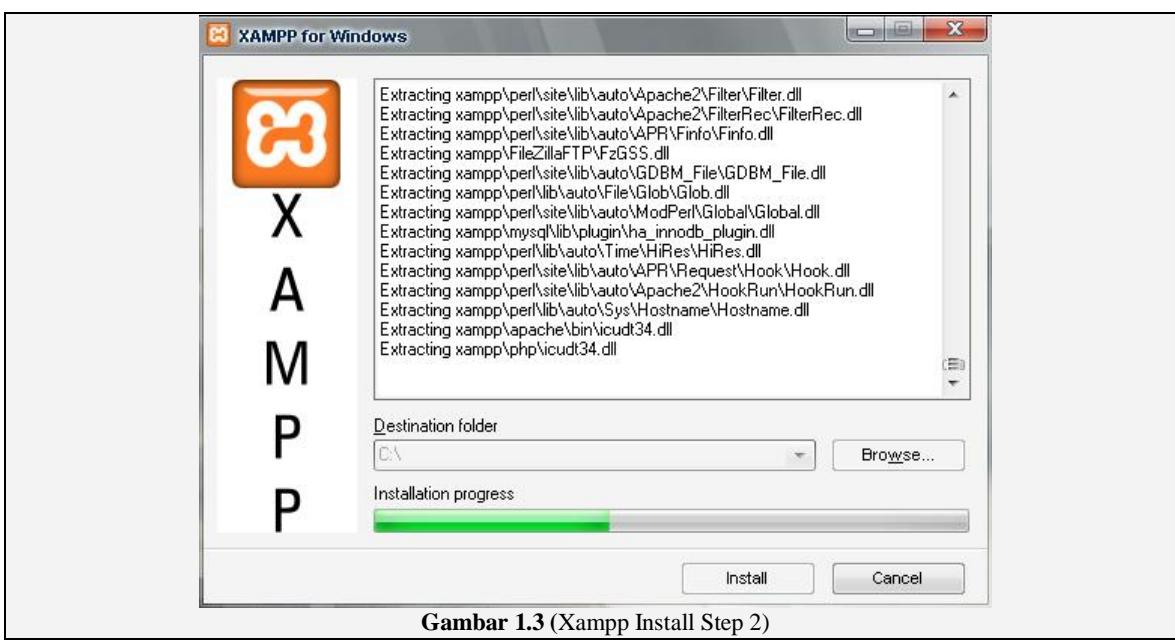
Langkah 2 :

Nah kalau sudah melihat button install, tangan kita sudah gatal-gatal ingin segera mengkliknya, biarkan seperti itu saja, kemudian klik install. Sehingga nanti xamppnya akan terinstall secara otomatis di **c:\xampp**



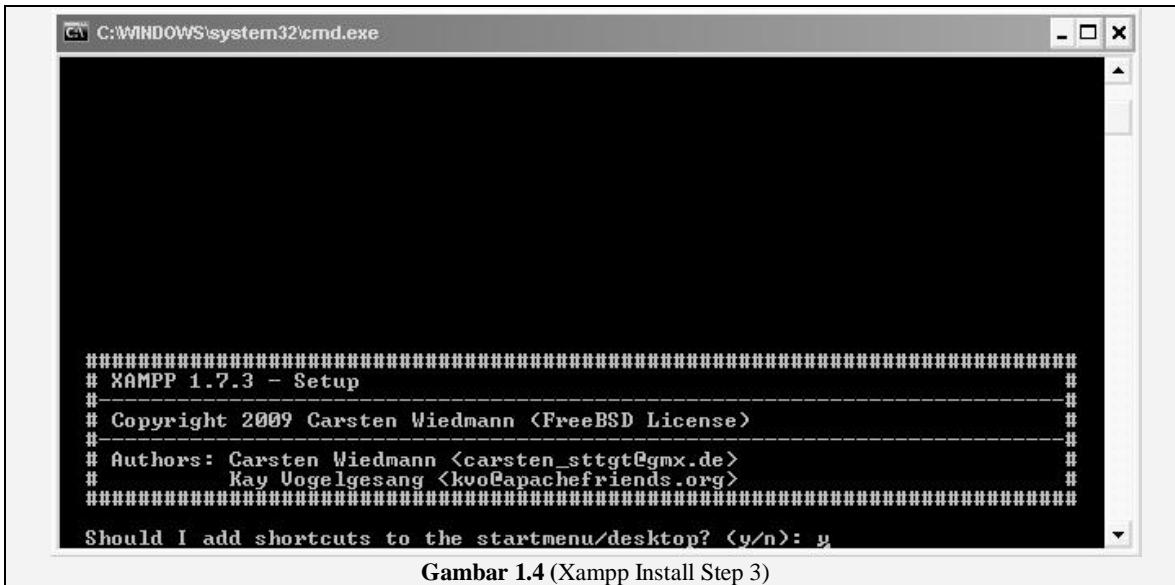
Langkah 3 :

Tunggu sampai progress bar (prosesnya) beres...



Langkah 4 :

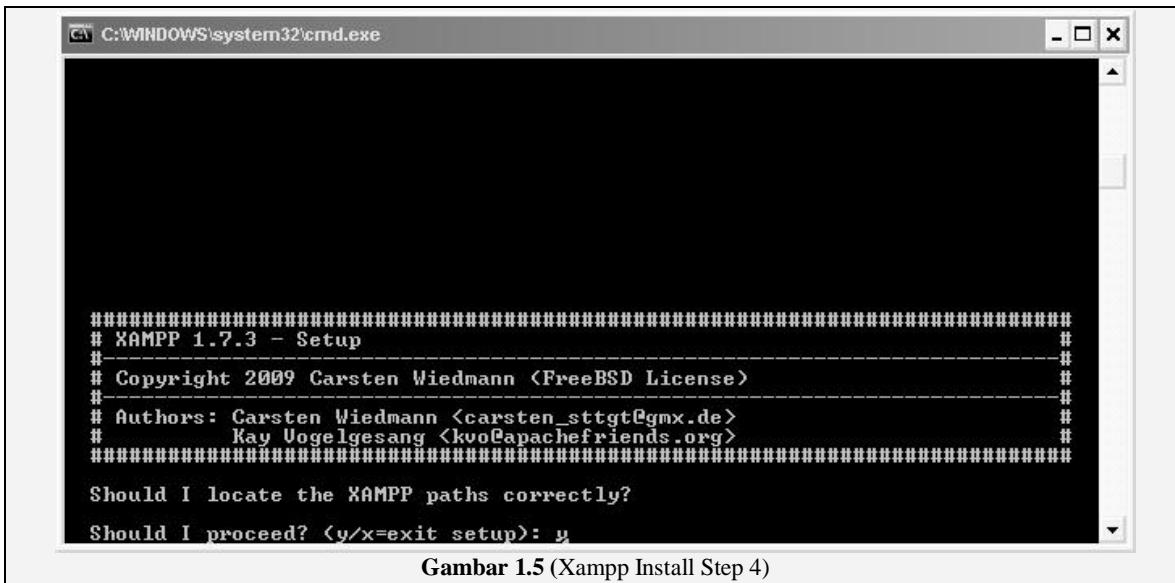
Apakah anda ingin shortcut xampp control panel di letakkan di desktop ataupun start menu ? ketik y saja untuk yes kemudian enter.



Gambar 1.4 (Xampp Install Step 3)

Langkah 5 :

Pertanyaan ini untuk memperbaiki direktori install, apakah sudah benar letak direktorinya? Ketik y saja kemudian enter.

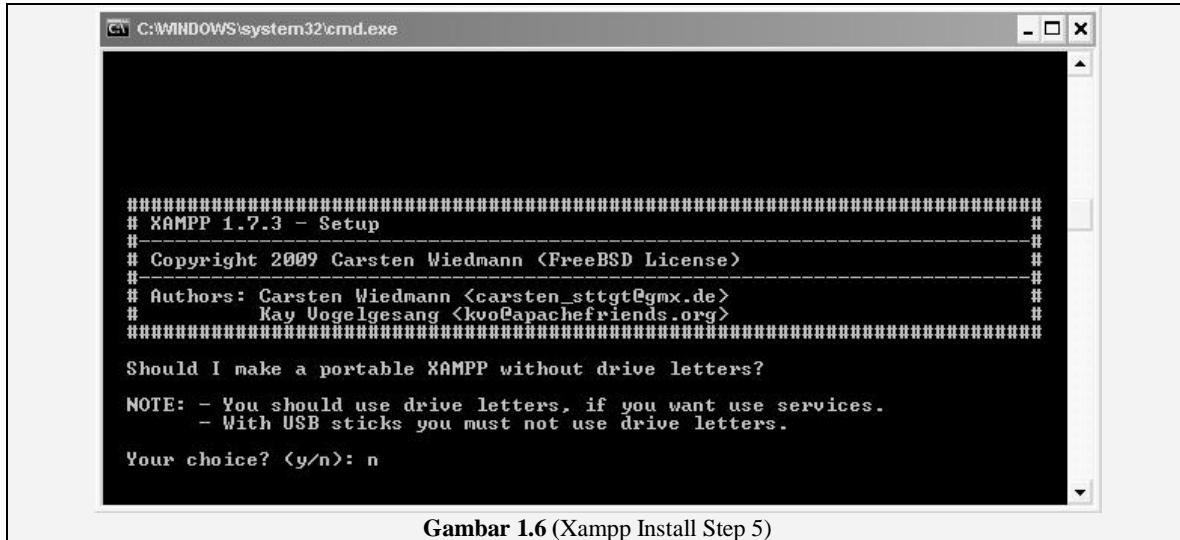


Gambar 1.5 (Xampp Install Step 4)

Langkah 6 :

Selanjutnya akan ada pertanyaan, apakah anda ingin memasukkan xampp kedalam flashdisk, sehingga xampp anda bisa digunakan secara portable, artinya anda dapat menjalankan secara portable, tidak perlu install lagi jika di jalankan dikomputer lain. Bagi yang sudah mengerti diharapkan diam saja, penulis sedang menjelaskan ke orang yang belum mengerti. Mohon bersabar. ☺.

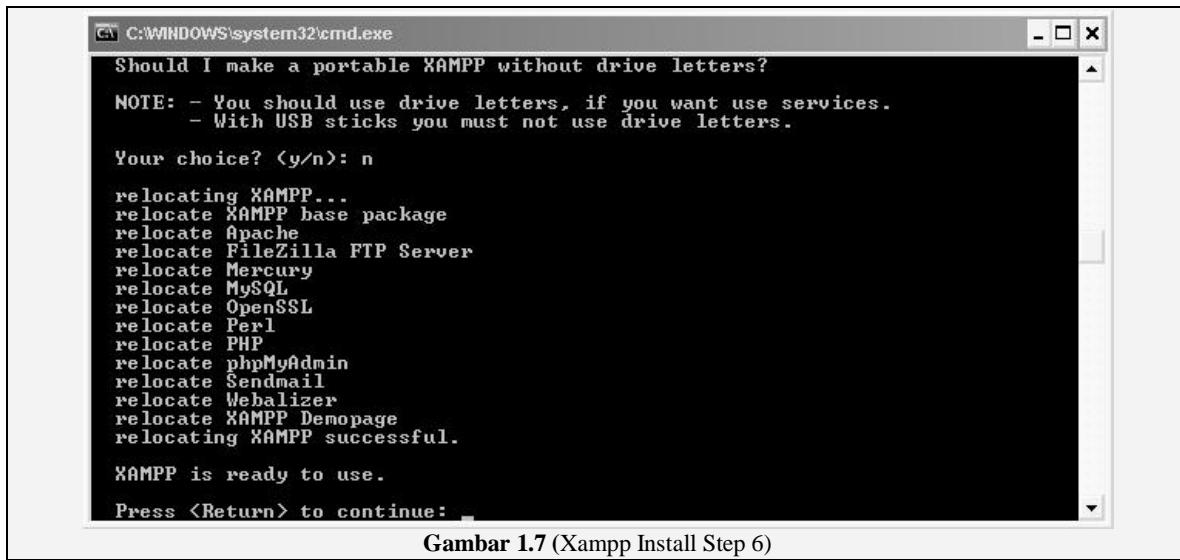
Ketik n saja untuk tidak, kemudian enter. Lalu diamkan sejenak.



Gambar 1.6 (Xampp Install Step 5)

Langkah 7 :

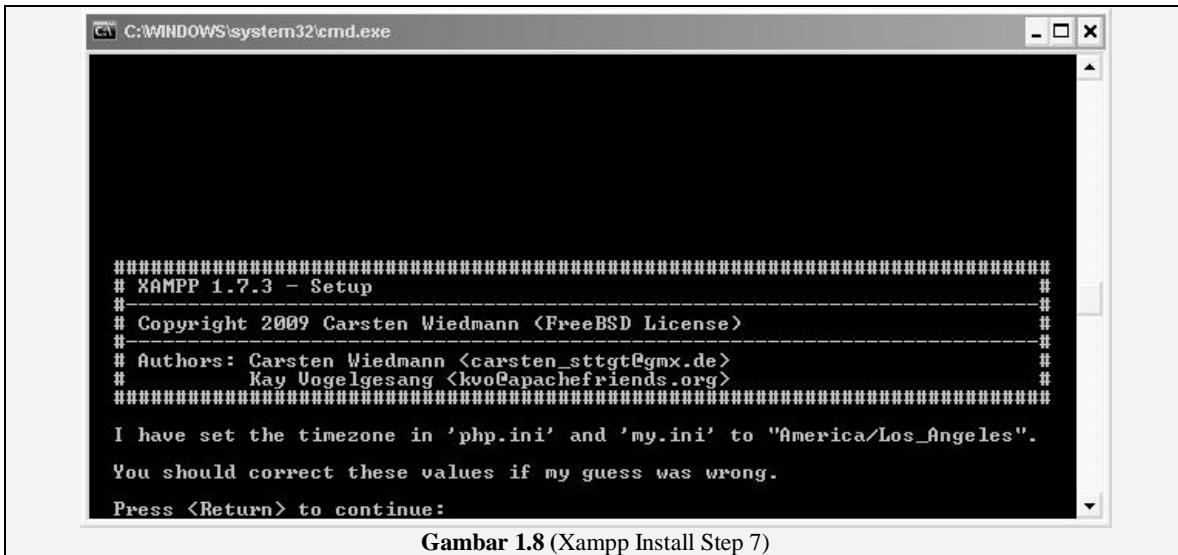
Lalu enter saja.



Gambar 1.7 (Xampp Install Step 6)

Langkah 8 :

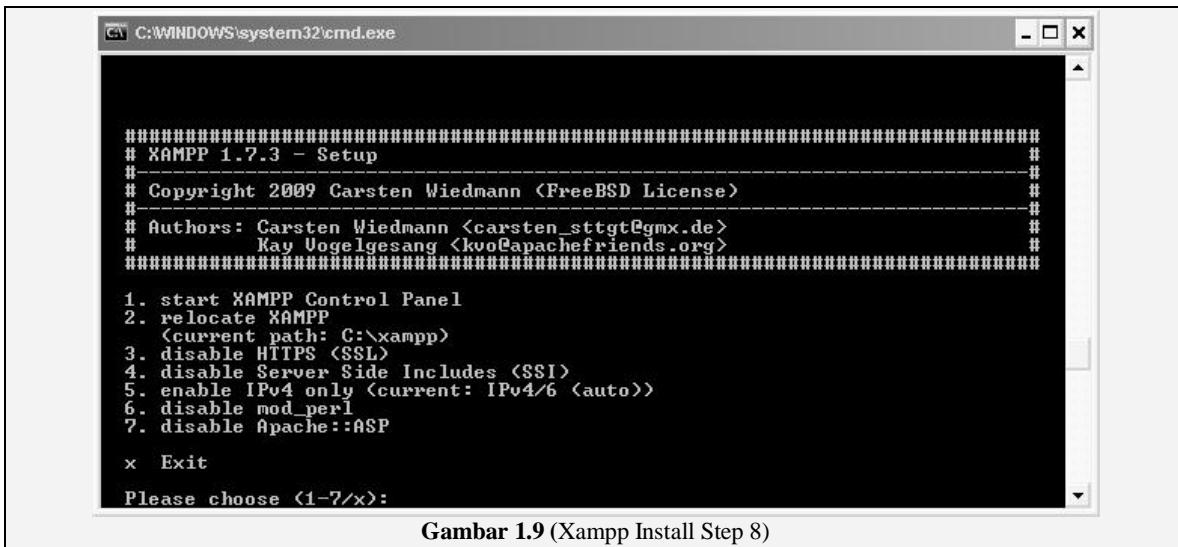
Lagi-lagi pertanyaan... Ck.. ck... ck.. Sabar... sabar... ☺ Silahkan enter saja.



Gambar 1.8 (Xampp Install Step 7)

Langkah 9 :

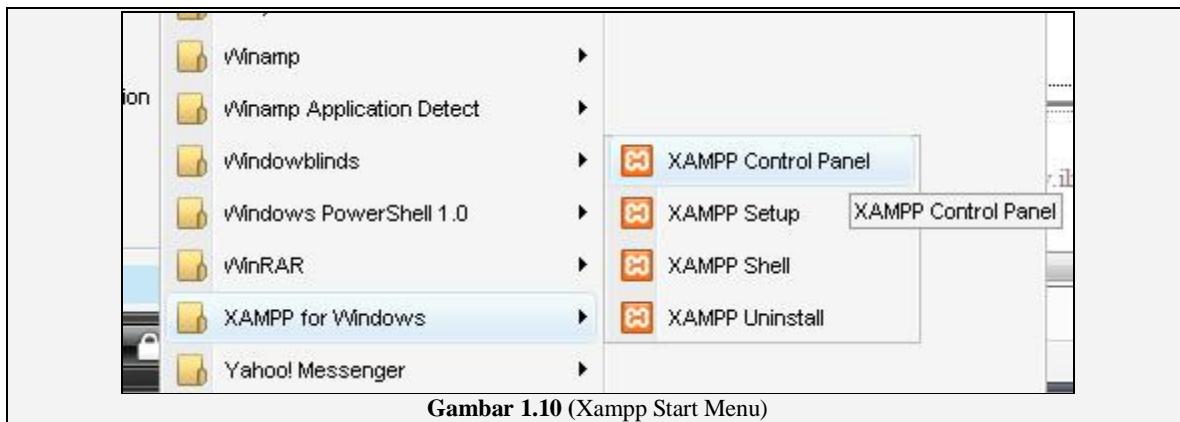
Oke, terakhir ada pertanyaan apakah anda ingin menjalankan pilihan yang disediakan?? Penulis menekan x kemudian enter untuk keluar dari install. So far, jika anda mengikuti proses ini dan tampilannya sama dengan yang penulis jabarkan disini berarti anda telah selesai menginstall xampp.



Gambar 1.9 (Xampp Install Step 8)

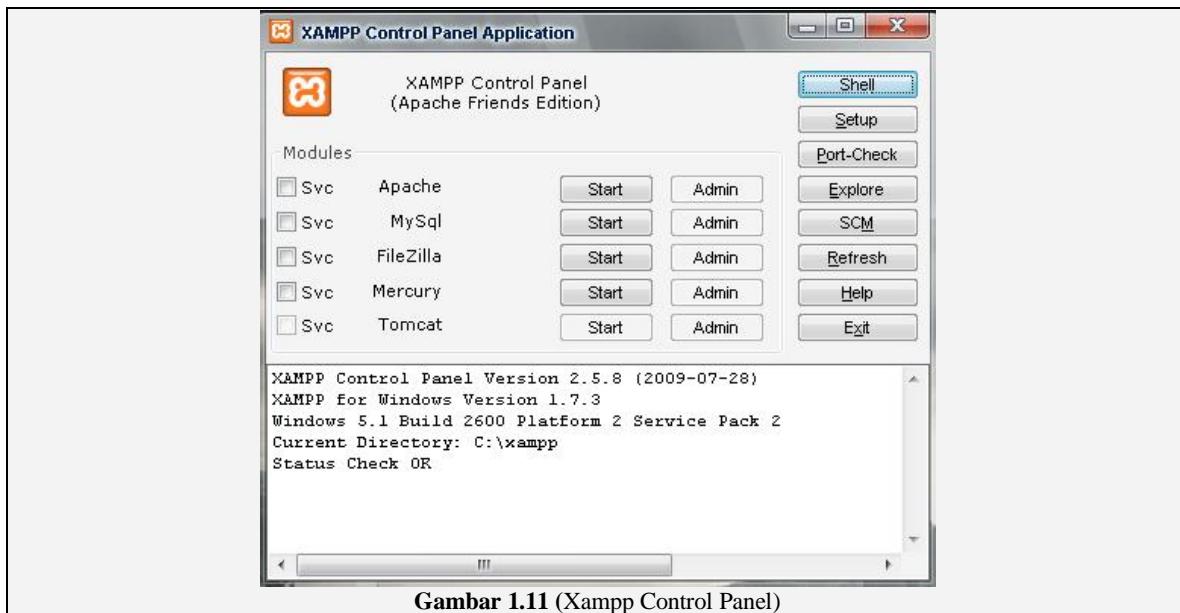
Selesai sudah installasi xampp versi 1.7.3, untuk menjalankan xampp anda bisa langsung double klik Xampp Control Panel di desktop.

Anda juga bisa mencarinya di start menu | xampp for windows kemudian klik xampp control panel.



Gambar 1.10 (Xampp Start Menu)

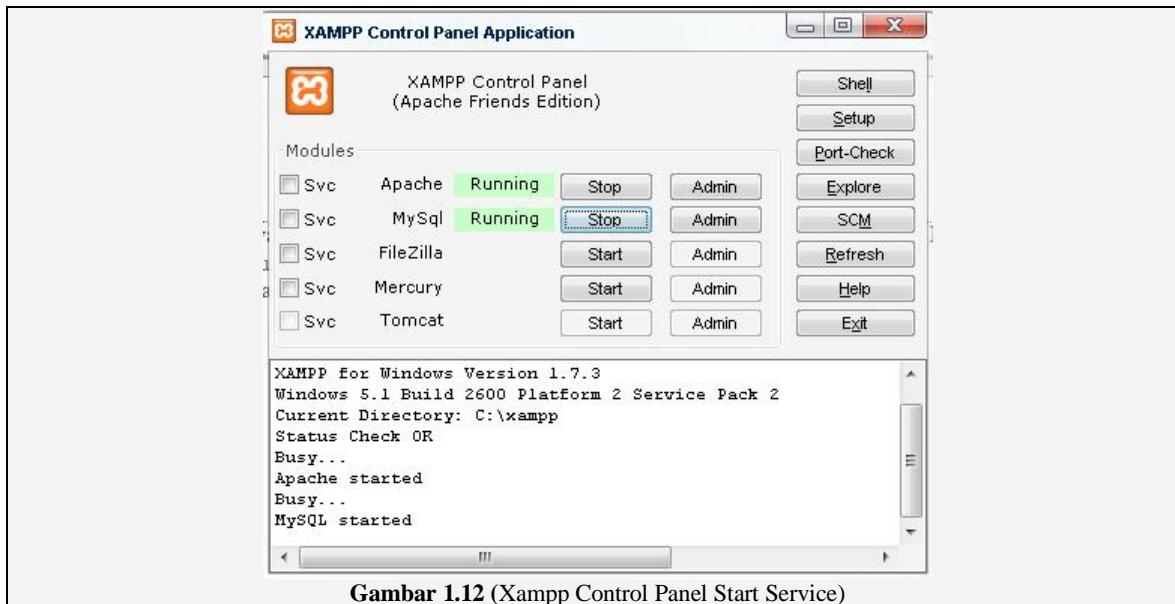
Kemudian nanti akan tampil windows seperti ini :



Gambar 1.11 (Xampp Control Panel)

Mungkin anda bertanya-tanya, apa yang harus saya klik? Dalam mempelajari PHP dan MySQL kita cuma butuh dua services yang berjalan. Pertama adalah Apache, dan kedua Adalah MySQL. Apache inilah yang nantinya dapat menjalankan script PHP, dan MySQL adalah service yang digunakan untuk dapat membuat database mysql.

Oke yang perlu diklik adalah start di baris Apache dan Start di baris MySQL, sisanya digunakan untuk FTP, Email, dan bahasa yang lain selain PHP. Oke? Silahkan klik start apache dan start mysql. Sehingga hasilnya menjadi seperti ini :



Jika sudah seperti ini, artinya anda telah berhasil menjalankan service apache dan mysql. Untuk mengetes apakah sudah benar-benar berhasil, anda bisa membuka browser internet favorit anda, di sini penulis menggunakan browser internet mozilla firefox. Kemudian ketikkan <http://localhost> di addressbar, kemudian enter. Apabila anda melihat tampilan yang sama seperti dibawah ini, anda sudah bisa bermain-main dengan PHP.

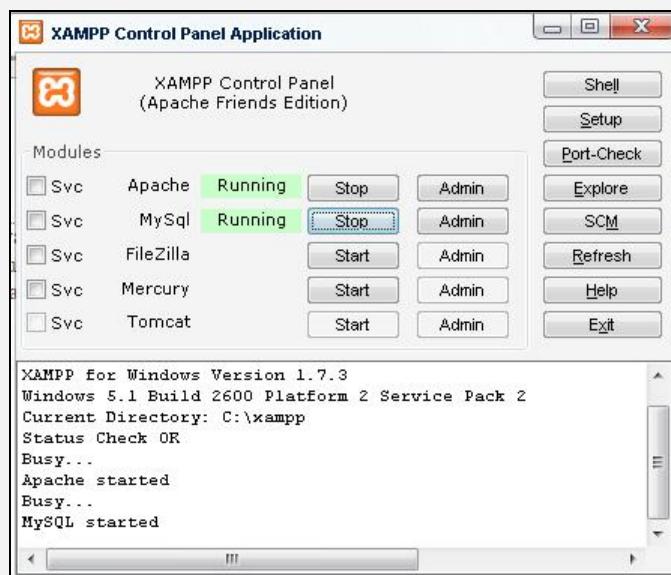


Bab 2:

Bekerja dengan PHP

2.1 Struktur Syntax PHP

Oke sebelum bekerja dengan php, pada bab sebelumnya kita sudah berhasil menginstall xampp, namun kita belum dapat menjalankan skrip phpnya dan juga belum dapat menikmati database mysqlnya. Untuk dapat menjalankan keduanya kita cukup, cari xampp-control panel di start menu, dan kita klik tombol start pada baris apache dan mysql, hingga nanti anda label hijau dengan tulisan running seperti ini :



Gambar 2.1 (Xampp Control Panel)

ini tandanya kita sudah dapat bekerja dengan mysql maupun php.

Kita akan coba masuk kedalam pembahasan, bekerja dengan PHP. Sebelumnya kita telah praktikan bagaimana menginstall xampp agar PHP dan MySQL dapat dijalankan di mesin lokal. Sekarang saatnya kita akan coba mengetahui struktur dari syntax PHP. Saya asumsikan anda sudah menguasai HTML.

Ya, PHP sama dengan HTML, untuk dapat menjalankan scriptnya maka anda harus mengetikkan syntax-syntaxnya ke dalam satu buah file dengan menggunakan text-editor seperti notepad, atau text editor kesukaan anda, kemudian meletakkan file tersebut pada suatu direktori khusus yakni defaultnya adalah di :

```
c:\xampp\htdocs\
```

Kemudian untuk dapat menjalankan skrip tersebut gunakanlah browser internet seperti mozilla firefox. Dengan mengisi alamat di address bar seperti ini misalnya :

```
http://localhost/direktori/file.php
```

saya akan jelaskan nanti arti dari alamat tersebut.

2.1.2 Struktur halaman PHP 1

Struktur halaman php bentuknya seperti ini :

```
<?php  
// syntax php di letakkan di sini ...  
// kode php di akhiri dengan ';'  
?>
```

Untuk mengetikkan baris code php, maka anda perlu mengapitnya dengan tag <?php dan di akhir dengan ?> karena di luar dari itu php engine tidak menganggapnya sebagai script php, melainkan hanya html biasa. Sebagai contoh kita akan menggunakan php, untuk menampilkan kalimat "Hallo dunia PHP?!" kita cukup membuka notepad, lalu ketikan seperti ini di notepad :

```
<?php  
echo "Hallo dunia PHP?!" ;  
?>
```

oke, setelah mengetik semua, silahkan save di :

```
c:\xampp\htdocs\
```

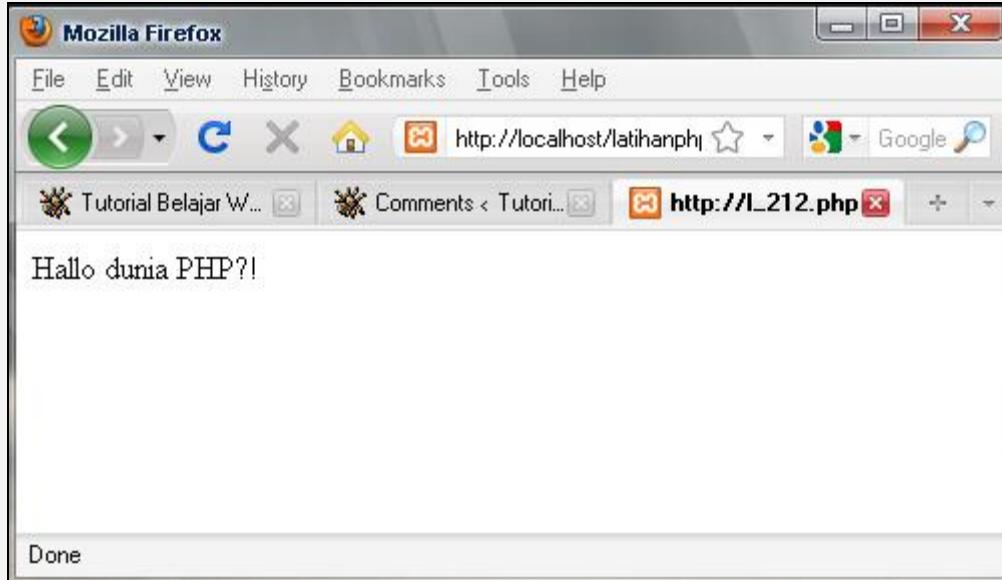
buat folder baru dengan nama latihanphp, lalu save dengan nama **latihan1.php**, sehingga nanti struktur pathnya adalah :

c:\xampp\htdocs\latihanphp\latihan1.php

untuk menjalankan script tersebut anda cukup mengetikkan di browser :

<http://localhost/latihanphp/latihan1.php>

hasil di browser adalah seperti ini :



Gambar 2.2 (Hallo PHP?)

ini terjadi karena direktori default apache untuk dapat menjalankan php adalah c:\xampp\htdocs, maka untuk dapat membuka file latihan1.php, kita cukup ketikkan alamat :

<http://localhost/latihanphp/latihan1.php>

Mudah ya? Mudah. Nggak ada yang sulit kan? nggak ada dong, iya kan??? semangaat! Anda dapat merubah direktori default dari apache dengan mengkonfigurasi ulang file httpd.conf yang terletak di :

C:\xampp\apache\conf\httpd.conf

2.1.3 Cara Penulisan Coding PHP yang baik

Oke, kita lanjut, agar ketika melakukan coding dengan PHP dapat terbaca dengan mudah, dan algoritmanya terbaca dengan jelas, maka dalam menyusun coding harus diatur sedemikian rupa.

Contoh penulisan coding yang buruk seperti adalah seperti list code di bawah ini. Ini cuma, contoh saja, mungkin anda belum mengerti arti dari baris-baris code tersebut, saya cuma memberikan contoh saja.

```
<?php
// file latihan_213.php
if ($_POST[ "nama" ] == "Anggie" )
{
echo "<p>Hei $_POST[ 'nama' ]</p>";
}
else
{
echo "<h2>Anda bukan Anggie!</h2>";
}

// masih dengan contoh yang sama,
// namun dengan cara penulisan yang berbeda

if ($_POST[ "nama" ] == "Anggie" )
{
echo "<p>" ;
echo "Hei ";
echo $_POST[ 'nama' ];
echo "</p>";
}
else
{
echo "<h2>" ;
echo "Anda bukan Anggie!";
echo "</h2>";
}
?>
```

Script ini sulit untuk dibaca, karena jika ada programmer lain atau mungkin anda yang bertahun-tahun liburan kemudian ingin membaca kode-kode yang anda buat tersebut tentunya akan mengalami kesulitan membedakan fungsi-fungsi dari baris

kodenya. Bayangkan jika seorang programmer mengetik kode-kode ratusan bahkan ribuan kode dengan cara penulisan seperti itu ??? Mungkin programer yang membacanya akan mual-mual. 8-}.

Untuk saat ini karena kita baru belajar bahasa pemrograman php, maka belum terasa kesusahannya jika tidak menulis kode dengan cara yang baik.

Saya mengibaratkan penulisan kode yang baik itu seperti seorang penulis mengarang sebuah cerita. Seorang penulis yang baik itu dapat mempermudah si pembaca dalam membaca tulisan karangan si penulis, mulai dari tanda baca, tahu letak berhenti suatu kalimat, paragraf, benar-benar memudahkan, beda dengan penulis yang salah urat, saya mengatakannya salah urat karena, penulis ini membuat cerita tapi tidak mempedulikan tanda baca, ini artinya penyiksaan dini bagi para pembacanya. Ya, contoh coding yang baik adalah seperti ini :

```
<?php
if ($_POST[ "nama" ] == "Anggie")
{
    echo "<p>Hei $_POST[ 'nama' ]</p>";
}
else
{
    echo "<h2>Anda bukan Anggie!</h2>";
}

// masih dengan contoh yang sama,
// namun dengan cara penulisan yang berbeda

if ($_POST[ "nama" ] == "Anggie")
{
    echo "<p>";
    echo "Hei ";
    echo $_POST[ 'nama' ];
    echo "</p>";
}

else
{
    echo "<h2>";
    echo "Anda bukan Anggie!";
    echo "</h2>";
}
?>
```

2.2 Mulai Membuat Halaman PHP

Pastikan xampp control panel berjalan, dan lihat kondisi Apache dan MySQL nya apakah sudah running ?? Jika iya, kita akan masuk ke dalam direktori :

```
c:\xampp\htdocs\latihanphp
```

Di direktori ini kita akan membuat satu buah file latihan2.php.

klik kanan -> new -> text document -> rename menjadi **latihan2.php**

kemudian buka file tersebut, kita akan mengetikkan beberapa baris kode. Sebagai awal belajar php. Kita ketikkan seperti ini :

```
<?php  
echo "Saya Belajar PHP";  
?>
```

kemudian simpan... dengan menekan ctrl + s di keyboard, lalu buka mozilla firefox. Kemudian arahkan alamatnya ke **http://localhost/latihanphp/latihan2.php**. Jika tampilannya adalah :

```
Saya Belajar PHP
```

berarti anda telah berhasil mengawali belajar php dengan sukses. Selamat!

2.3 Integrasi HTML dan PHP

Selain berdiri sendiri, PHP pun dapat di integrasikan dengan HTML. Saya asumsikan anda sudah mengerti HTML sebelumnya. Kita akan coba membuat sebuah halaman yang didalamnya terdapat script PHP dan HTML.

Masuk ke dalam direktori `c:\xampp\htdocs\latihanphp\` buat sebuah file dengan nama `latihan3.php`, kemudian ketikan baris code seperti ini...

```
<!-- file latihan3.php -->
<html>
<head>Latihan 3</head>
<body>
<?php
echo "Ini Script PHP di dalam tag HTML";
?>
</body>
</html>
```

untuk dapat melihat hasilnya, buka mozilla firefox, dan ketikkan alamatnya :

```
http://localhost/latihanphp/latihan3.php
```

ya, kita lihat view source code htmlnya, dengan cara mengklik kanan browser lalu pilih View Page Source, di sini kita tidak lagi melihat script php-nya, yang ada hanyalah HTML saja, ini terjadi karena PHP engine/processor mengeksekusi file php di server, dan menghasilkan html dan mengirimkan file htmlnya kepada user, sehingga yang terlihat adalah htmlnya saja, bukan phpnya.

2.4 Menggunakan Constant dan Variable

Kita akan mendalami PHP lebih lanjut. PHP memiliki dua jenis nilai, yakni Constant dan Variable. Keduanya sangat berpengaruh kedalam pemrograman PHP itu sendiri, karena jika tidak ada kedua nilai ini maka PHP hanyalah barisan code saja tanpa nilai, atau dengan kata lain, tidak berfungsi sebagai apa-apa.

Kita akan bahas secara lengkap dalam bagian ini kedua nilai yang saya sebutkan tadi.

2.4.1 Menggunakan Constant

Apa itu konstan ?? konstan diartikan sebagai suatu nilai yang tidak bisa di modifikasi lagi. Bersifat mutlak, dan tidak dapat di rubah. biasanya ini digunakan untuk bilangan yang tetap, misalkan untuk jari-jari suatu lingkaran, misalnya. Atau mungkin tanggal lahir seseorang.

Misalkan :

```
Pi = 3,141592
```

dalam php itu untuk mendefiniskan kontan Pi adalah sebagai berikut :

```
define ("Pi", 3.141592);
```

apabila kita menginginkan untuk menampilkan isi dari Pi, caranya adalah seperti ini.

```
<?php
define ("Pi", 3.141592);
echo Pi;
?>
```

Oke, misalkan kita ingin mengintegrasikan nya dengan html, kita buat satu file di **c:\xampp\htdocs\latihanphp** bernama **latihan4.php**

kemudian ketikkan isinya seperti ini : ...

```
<!-- file latihan4.php -->
<html>
<head>Latihan 4</head>
<body>
<?php
define ("Pi", 3.141592);
echo Pi;
?>
</body>
</html>
```

kita buka di mozilla firefox, kemudian arahkan link :

`http://localhost/latihanphp/latihan4.php`

hasilnya adalah seperti ini :

`3.141592`

membuat konstan sangat mudah sekali bukan?

2.4.2 Menggunakan Variable

Variable, merupakan suatu nilai yang masih dapat di ubah, dengan cara di isi ulang, berbeda dengan konstan yang tidak dapat di rubah dan bersifat mutlak. Coba kita buat satu buah file di `c:\xampp\htdocs\latihanphp\` bernama `latihan5.php`. Kemudian kita ketikkan baris seperti ini ke dalam file tersebut :

```
<?php  
$tahun = 2010; // ini adalah proses assignment  
echo $tahun ;  
  
$tahun = 2012;  
echo $tahun ;  
?>
```

kemudian kita eksekusi filenya dengan browser mozilla firefox dan menulis alamatnya seperti ini :

`http://localhost/latihanphp/latihan5.php`

Terlihat nilai 2012 menimpa nilai yang sebelumnya berbeda dengan constant.

Catatan :

“Kita lihat `$tahun = 2010;`
yang dinamakan variable adalah `$tahun`
sedangkan nilainya adalah 2010
sehingga variable `$tahun` memiliki nilai 2010”

Ada beberapa tipe data untuk variable, variable yang baru kita praktikan itu bertipe integer. Ada 3 jenis tipe data induk, yakni integer, float, dan string.

Catatan :

Tipe Integer itu tipe data yang berisi bilangan bulat, 0, 1, 2, 3, dan seterusnya...

Tipe Float itu tipe data yang berisi bilang pecahan atau desimal, seperti 1,35 atau 7,5 dan lain sebagainya...

Tipe String itu tipe data yang berisi kumpulan karakter, angka, huruf, simbol, misalnya seperti "Susu Ultra Jaya", atau nama jalan "Jl. Warung Nangka 2, Gang Duren Mateng"

Cara Penulisan Variable :

Dalam menulis variable ada hal-hal yang harus diperhatikan, dan jika tidak diikuti maka variable tersebut tidak akan menjadi variable, tapi bilangan lain atau mungkin error :

\$variable	→ harus di awali oleh huruf atau _, tidak boleh memuat spasi tidak boleh memuat spesial karakter (~!@#\$% dan lain lain)
\$892	→ ini contoh yang salah dalam penulisan variable
\$ini bukan variable	→ ini contoh yang salah dalam penulisan variable
\$_variable1	→ ini adalah contoh yang benar dalam penulisan variable

2.4.3 Menggunakan Operator

Operator membuat suatu nilai menjadi penting. Ada beberapa jenis operator yang berguna dalam pemrograman menggunakan PHP. Ada operator matematika, assignment, perbandingan, logika, increment, decrement.

Operator matematika

Contohnya adalah seperti ini :

```
<?php
// contoh operator matematika
$pengurangan = 3.033.032.020 - 3.033.031.924;
$pertambahan = 3.033.032.020 + 3.033.031.924;
$perkalian = 3.033.032.020 * 3.033.031.924;
$pembagian = 3.033.032.020 / 3.033.031.924;
$hasilbagi = 200 % 4 ;

echo "3.033.032.020 - 3.033.031.924 = $pengurangan" ;
```

```

echo "3.033.032.020 + 3.033.031.924 = $pertambahan" ;
echo "3.033.032.020 * 3.033.031.924 = $perkalian" ;
echo "3.033.032.020 / 3.033.031.924 = $pembagian" ;
echo "200 % 4 = $hasilbagi";

// pengurangan di wakili tanda '-'
// penjumlahan di wakili tanda '+'
// perkalian di wakili tanda '**'
// pembagian di wakili dengan tanda '/'
// sisa hasil bagi di wakili dengan tanda '%'
?>

```

Operator matematika ini layaknya anda berhitung menggunakan matematika, mulai dari pengurangan, penjumlahan, perkalian, pembagian.

Operator assignment

Operator assignment merupakan operator yang berguna untuk memasukkan nilai kedalam suatu variable. Dan jangan lupa, setiap baris code harus di tutup dengan ';' atau script anda akan error.

Coba kita praktikan :

```

<?php
// contoh operator assignment
$panjang = 10; // contoh assignment
$lebar = 8; // contoh assignment
$luas = $panjang * $lebar ; // contoh assignment juga

// tanda '=' mewakili proses assignment,
// ada beberapa tanda lagi yang dapat
// melakukan assignment,
// assignment penjumlahan diwakili tanda '+=' contohnya
$tambahdua += 2;
echo $tambahdua."<br>" ;

// hasilnya 2
// hal ini sama dengan pernyatan :
// $tambahdua = $tambahdua + 2;
// assignment pengurangan diwakili tanda '==' contohnya

$clip = 22;
$clip -= 2; // sama dengan $clip = $clip - 2;
echo $clip;
?>

```

hal ini juga sama dengan perkalian di wakili tanda '*' dan pembagian '/'

contoh yang lebih lengkapnya adalah seperti ini :

```
<?php  
$a = 50;  
$b = 40;  
$c = 30;  
$d = 20;  
  
echo "Hasil dari 50+=2 adalah " . $a+=2 ;  
echo "Hasil dari 40-=2 adalah " . $b-=2 ;  
echo "Hasil dari 30*=2 adalah " . $c*=2 ;  
echo "Hasil dari 20/=2 adalah " , $d/=2 ;  
?>
```

kemudian save di **c:\xampp\htdocs\latihanphp\latihan7.php** lalu jalankan di mozilla firefox, atau browser lain. kita lihat hasilnya.

logikanya adalah sebagai berikut, kita ambil salah satu, misalkan :

```
$a+=2
```

ini berarti sama saja dengan :

```
$a = $a + 2;
```

karena a bernilai 50, maka tambahkan 2 kepada 50, sehingga **50 + 2 = 52** mudah bukan. ? Begitu juga untuk pengurangan, perkalian, dan pembagian. Perlu di catat di sini, ada bentuk baris seperti ini :

```
echo "Hasil dari 50+=2 adalah " . $a+=2 ;
```

dengan tanda titik, tanda titik memiliki arti melanjutkan kalimat. anda juga bisa menggantikan titik dengan koma. Sebagai contoh apabila kita ingin menulis kalimat "Saya belajar PHP" kedalam 3 bagian, kita dapat menulisnya seperti ini

```
echo "Saya" . "Belajar" . "PHP" ;
```

atau juga bisa seperti ini :

```
echo "Saya" , "Belajar" , "PHP" ;
```

sehingga untuk menampilkan angka 5 yang berada dalam variable \$a, bisa seperti ini :

```
echo "Nilai a adalah $a";
```

bisa juga seperti ini :

```
echo "Nilai a adalah". $a ;
```

atau bisa juga seperti ini :

```
echo "Nilai a adalah", $a ;
```

mudah bukan ??

Selanjutnya adalah **operator perbandingan**, khusus untuk **operator perbandingan** tidak dapat digunakan standonline, artinya tidak bisa bekerja jika tidak di integrasikan dengan kondisional / percabangan , oleh karena itu saya akan membahasnya berbarengan dengan kondisional if / else. Begitu juga dengan operator logika, saya akan jelaskan di bagian kondisional, sedangkan untuk operator increment, dan decrement akan dijelaskan di bagian pengulangan.

2.5 Menggunakan Argumen If dan Else

2.5.1 Menggunakan Argumen IF

Argumen if atau keputusan if sangat sering digunakan oleh para programmer php, alasannya adalah mudah dan sederhana. Penggunaan argumen if ini biasanya dilakukan ketika akan membandingkan suatu nilai, atau membandingkan suatu kondisi.

Dalam dunia nyata membandingkan kondisi itu seperti pengandaian, contohnya adalah seperti ini :

```
jika cuaca cerah  
maka saya akan berangkat kuliah.
```

ini berarti dimana "jika cuaca cerah" itu adalah nilai atau kondisi dan

"maka saya akan berangkat kuliah" ini di sebut dengan pernyataan atau statement dalam php. mudah ya?

cara penulisan argumen if dalam dunia php adalah sebagai berikut :

```
if kriteria {  
    pernyataan;  
}
```

kita coba implementasikannya :

Buat sebuah file bernama **latihan8.php** dan simpan di c:\xampp\htdocs\latihanphp, kemudian isi filenya seperti ini :

```
<?php  
$cuaca = "cerah";  
if ($cuaca == "cerah")  
{  
    echo "Saya akan berangkat kuliah!";  
}  
?>
```

Lihat hasilnya di browser...

```
Saya akan berangkat kuliah!
```

Bagaimana???

Bab2: Bekerja dengan PHP

Oleh : Loka Dwiartara | <http://www.ilmuwebsite.com>

Kembali ke coding, dalam baris kode urutan ke 3, ada kalimat :

```
if ($cuaca == "cerah")
```

tanda '==' ini merupakan operator perbandingan, yang membandingkan isi variable \$cuaca dengan nilai "cerah", jika bernilai benar, maka akan ditampilkan pernyataan di bawahnya.

perlu dicatat sekali lagi, jika :

```
$cuaca = "cerah" ini berarti memasukkan nilai "cerah" kedalam variable  
$cuaca,
```

Sedangkan :

```
if ($cuaca == "cerah") ini berarti menanyakan/membandingkan, apakah  
nilai variable $cuaca itu berisi "cerah"??
```

jika benar, maka pernyataan yang berada di dalam tanda {} akan ditampilkan, jika tidak maka php akan berhenti mengeksekusi filenya. Contoh yang lain adalah sebagai berikut :

```
<?php  
$jarak = 40;  
  
if ($jarak <= 40)  
{  
    echo "Jalan kaki saja";  
}  
  
if ($jarak >= 40)  
{  
    echo "Naik Motor";  
}  
  
if ($jarak != 40)  
{  
    echo "Diam ditempat";  
}  
?>
```

Hasilnya nanti adalah :

```
Jalan Kaki Saja Naik Motor
```

2.5.2 Menggunakan Argumen if dan else secara bersamaan

Argumen if dan else apabila digunakan secara bersamaan maka akan dapat menghasilkan dua statement. Jika if itu hanya dapat menghasilkan satu statement saja, maka if dan else akan menghasilkan 2 opsi statement pada satu kondisi. Dalam kehidupan nyata jika anda mengalami suatu kondisi maka akan terdapat beberapa opsi, contohnya adalah sebagai berikut :

```
jika cuaca cerah
maka saya akan berangkat kuliah

jika tidak
maka saya akan membuat mie ramen
```

dan, apabila diaplikasikan ke dalam bahasa php. Mudahnya adalah sebagai berikut :

```
<?php
$cuaca = "mendung";
if ($cuaca == "cerah") // jika cuaca cerah
{
    echo "Saya akan berangkat kuliah";
}

else
{
    echo "Saya akan membuat mie ramen";
}
?>
```

contoh yang lainnya :

```
<?php
$cuaca = "mendung";
if ($cuaca != "cerah") // jika cuaca tidak cerah
{
    echo "Saya akan membuat mie ramen";
}

else
{
    echo "Saya akan berangkat kuliah";
}
?>
```

Mudah bukan? Kemudahan inilah yang membuat para web developer menjawab "PHP itu sangat manusiawi" ketika saya menanyakan tentang bahasa apa yang mereka sukai.

2.5.3 Menggunakan Argumen if, else if, dan else

Jika pada argumen if dan else itu dapat menggunakan dua kondisi, yakni kondisi jika benar, dan jika salah, pada argumen if, else if, dan else, anda dapat menggunakan kondisi yang tidak terbatas. dan masing-masing kondisi/kriteria memiliki masing-masing pernyataan/statement. Dalam dunia sehari-hari argumen if, else if, dan else memiliki contoh seperti ini :

jika mendung

maka bawa payung

jika cerah

maka saya akan berangkat kuliah dengan jalan kaki meskipun jaraknya 20 km.

jika banjir

maka bawa perahu sendiri dari rumah

dan begitu seterusnya, tidak ada batasnya...

dalam dunia php, contoh implementasinya adalah sebagai berikut, jangan lupa untuk memperhatikan tanda {} dan ; nya

```
<?php
$nilai = "A";

if($nilai == "A")
{
    echo "Bagusss... baguss...";
}

else if ($nilai == "B")
{
    echo "Baik...";
}

else if ($nilai == "C")
{
    echo "Cukup...";
```

```
}

else if ($nilai == "D")
{
    echo "Kurang...";
}
// dan seterus-seterusnya ...
?>
```

silahkan save di c:\xampp\htdocs\latihanphp\ . Kita lihat di browser mozilla firefox hasilnya adalah seperti ini :

```
Bagus... bagus...
```

di dalam penggunaan argumen if, else if, dan else terlihat begitu tidak terbatas jika di aplikasikan kedalam kasus yang lain.

2.6 Menggunakan pemilihan switch

Switch sebetulnya sama saja dengan argumen if, else if, dan else, cuma sedikit berbeda dalam penggunaannya terlebih pada syntaxnya. Jika untuk memutuskan kondisi dalam if, else if, dan else itu menggunakan {} namun dalam switch untuk memberhentikan keputusan digunakanlah fungsi break; bentuk aslinya adalah seperti ini :

```
switch(variable){ // kita memilih variable yang akan dicocokkan
    case kriteria1: // kita menanyakan kriteria pertama itu apa
        pernyataan1; // kemudian pernyataannya apa ?
        break; // di akhir dengan break; begitu seterusnya, hingga kriteria habis

    case kriteria2:
        pernyataan2;
        break;
    // dan seterusnya dan seterusnya ...
}
```

di sini terlihat ketika kondisinya sesuai di wakili dengan code ‘case kriteria1:’ dengan titik dua ‘:’ di belakang setiap kriterianya. maka di tulis lah pernyataan di bawahnya itu apa?

contoh kasus dalam php nya sebagai berikut :

```
<?php
$nilai = "A";

switch($nilai)
{
    case "A":
        echo "Bagusss... baguss...";
        break;

    case "B":
        echo "Baik...";
        break;

    case "C":
        echo "Cukup...";
        break;
}
```

```
case "D":  
    echo "Kurang... ";  
    break;  
}  
?>
```

switch itu hanya digunakan pada satu jenis variable saja namun memiliki nilai yang berbeda. tetapi dalam if, else if, dan else dapat digunakan pada variable yang berbeda-beda dan nilai yang berbeda. switch ataupun if, else if, dan else tentunya dapat anda gunakan sesuai dengan kebutuhan anda nantinya.

2.7 Menggunakan Pengulangan

Dengan echo atau print, anda dapat menampilkan kalimat apapun, dan panjangnya pun tidak terbatas. Penggunaan echo atau print untuk mengetikan angka 1 - 3, misalnya seperti ini :

```
<?php
    echo "Angka 1<br>" ;
    echo "Angka 2<br>" ;
    echo "Angka 3<br>" ;
?>
```

Lihat hasilnya di browser anda. Bagaimana jadinya jika anda mendapatkan kasus untuk menulis angka hingga 100 baris atau 50jt baris? Apa masih menggunakan cara kuno tersebut? contoh yang lainnya bagaimana seorang programmer mengambil isi database kemudian menampilkannya semua isi database tersebut, apa masih menggunakan echo saja? Solusinya akan dibahas dalam pengulangan.

Terdapat 3 jenis pengulangan yang paling sering digunakan dalam php, pengulangan **bentuk for**, **bentuk while**, dan **bentuk do while**. saya akan membahas ketiganya sekarang.

2.7.1 Pengulangan For

For merupakan jenis pengulangan yang cukup sering digunakan, dan penggunaan pengulangan bentuk for ini tergantung kebutuhan. dan for ini bisa digunakan untuk contoh kasus sederhana yang saya sebutkan di awal tadi yaitu menampilkan angka 1 - 100 atau 1 - 50jt dengan cepat. For membuat semuanya menjadi sangat effisien. Bentuk umum for adalah seperti ini :

```
for(nilai awal; batas nilai; operator increment/decrement)
{
    pernyataan yang akan di proses
}
```

Contoh implementasinya seperti ini, menampilkan kalimat Angka 1 sampai 100 :

```
<?php
// file latihan_271.php
for($x=1;$x<=100;$x++)
{
    echo "Angka $x<br>" ;
```

```

    }
?>

```

Hasilnya :

```

1
2
3
4
...
100

```

Pengulangan ini akan terus menerus mengulang dan terus menerus dilakukan penambahan 1 hingga nilainya mencapai yang ditargetkan, yakni kurang sama dengan 100. Karena pengulangan terjadi dari angka 1 hingga 100 maka terdapat 100 proses.

Untuk operator increment yang lain adalah seperti ini.

```

<?php
for($x=1;$x<=100;$x+2)
{
    echo "Angka $x<br>";
}
?>

```

ini berarti dilakukan penambahan 2, sehingga nanti hasilnya seperti ini. (browser) dan untuk operator decrement contohnya seperti ini :

```

<?php
for($x=100;$x>=1;$x--)
{
    echo "Angka $x<br>";
}
?>

```

bedanya adalah pada operator perbandingan, jika pada pengulangan increment atau menaik bentuk operatornya adalah ' \leq ', namun dalam pengulangan decrement atau menurun bentuk operatornya adalah ' \geq ', apabila peraturan ini tidak dipatuhi maka hasilnya adalah error. Contoh lain pengulangan decrement :

```

<?php
for($x=100;$x>=1;$x-=2)
{

```

```

        echo "Angka $x<br>" ;
    }
?>

```

selain hanya php, anda juga dapat menggunakan pengulangan ini untuk menghasilkan html yang dinamis, contohnya seperti ini :

```

<html>
<head>
    <title>Pengulangan header</title>
</head>
<body>
    <?php
    for ($x=1;$x<=7;$x++)
    {
        echo "<font size=$x>Ukuran font $x</font><br>" ;
    }
    ?>
</body>
</html>

```

2.7.2 Pengulangan while

Pengulangan while ini bentuknya seperti ini :

```

Nilai awal
While (batas pengulangan)
{
    pernyataan;
    operator decrement/increment;
}

```

Pengulangan jenis ini tidak terlalu jauh berbeda dengan pengulangan bentuk for. hanya saja penempatan nilai awalnya, dan operator decrement/incrementnya berbeda letaknya. Implementasinya adalah seperti ini :

```

<?php
$x = 1;
while ($x <= 100)
{

```

```

        echo "Angka $x";
        $x++ // jika tidak ada operator increment atau decrement
              // maka proses akan
              // terus menerus terjadi tanpa berhenti
    }

?>

```

jadi jika kondisi belum sampai yang ditentukan dalam hal ini, jika variable \$x itu kurang dari 100 maka pengulangan akan terus dilakukan, dan penambahan selalu terjadi. namun ketika nilainya lebih dari 100 maka pengulangan akan berhenti. hampir sama dengan konsep pengulangan bentuk for.

2.7.3 Pengulangan do while

Bentuk pengulangan do while adalah seperti ini :

```

Nilai awal
do
{
    pernyataan;
    operator decrement/increment;
}
While (batas pengulangan)

```

contoh implementasinya :

```

<?php
$x = 1;
do
{
    echo "Angka $x";
    $x++
}
while ($x <= 100);
?>

```

hampir sama dengan bentuk pengulangan while, namun yang jadi perbedaan di sini adalah letak dari whilenya saja, dan diakhir dari pengulangan while di beri ; (semicolon).

coba kita buktikan :

```
<?php
// pengulangan jenis while
$x = 1;
while ($x <= 100)
{
    echo "Angka $x<br>";
    $x++;
}
echo "<strong>$x</strong>";
echo "<br><br>";

// pengulangan jenis do while
$x = 1;
do
{
    echo "Angka $x<br>";
    $x++;
}
while($x <= 100);
echo "<strong>$x</strong>";
?>
```

Di akhir setiap pengulangan saya memasukkan baris echo \$x ; ini cuma untuk membuktikan apakah benar nilainya sama, dan hasilnya ... sama. ini menandakan do while itu sama dengan while, namun implementasinya itu sesuai dengan kebutuhan anda.

Sampai sini, anda dapat memilih, lebih prefer atau lebih suka menggunakan pengulangan bentuk yang mana? for, while, atau do while. semua di tangan anda. dan gunakan secara efisien.

2.8 Menggunakan Include dan require untuk efisiensi Coding

Fungsi include dan require berfungsi untuk menjalankan banyak file secara berbarengan dengan cara memasukkan file lain kedalam suatu file tertentu. bentuk penulisannya adalah seperti ini :

```
include "namafile.php";
require "namafile.php";
```

Contoh implementasinya adalah seperti ini, kita akan membuat suatu argumen yang nilai variable nya itu diambil dari file yang lain. Pertama buat satu buah file dengan nama, variable.latihaninclude.php kemudian ketikkan di dalamnya seperti ini :

```
<?php
// file variable.latihaninclude.php
$cuaca = "hujan";
?>
```

kemudian kita akan mengambil variable \$cuaca tersebut untuk dimasukkan kedalam file yang lain, kita buat 1 buah file lagi, dengan nama, latihan_28.php, kemudian ketikkan di dalamnya seperti ini :

```
<?php
include "variable.latihaninclude.php";

if ($cuaca == "cerah")
{
    echo "Latihan bola di GOR";
}

else
{
    echo "Bikin mie ramen";
}
?>
```

hasilnya adalah :

```
Bikin mie ramen
```

Mengapa ? karena fungsi include lah yang dapat memanggil file lain untuk di

eksekusi bersama, namun perlu diperhatikan peletakkannya. jika di letakkan di akhir seperti ini maka akan error.

```
<?php
if ($cuaca == "cerah")
{
    echo "Latihan bola di GOR";
}

else
{
    echo "Bikin mie ramen";
}

include "variable.latihaninclude.php";
?>
```

oleh karena itu disesuaikan dengan penggunaannya, yakni diletakkan sebelum pengolahan file.

2.9 Menggunakan Fungsi untuk effisiensi Coding

Fungsi ini berisi sekumpulan perintah program, bisa berisi argumen if, pengulangan, variable, yang dapat menerima inputan berupa variable berisi nilai kemudian mengolah nilai tersebut sesuai dengan kebutuhan. Bentuk umum dalam membuat fungsi adalah seperti ini

```
function namafungsi(parameter1, parameter2, ... parameter n )
{
    pernyataan berupa perintah operasi program;
}
```

contoh implementasinya seperti ini :

```
<?php
// pertama kita terlebih dahulu mendefinisikan fungsinya,
// kemudian mengisi fungsi tersebut dengan sederetan perintah

function testing()
{
    echo "Contoh menggunakan Fungsi";
}

// untuk dapat menggunakan fungsi yang telah dibuat,
// perlu pemanggilan nama fungsinya.
testing();
?>
```

Menggunakan parameter dalam fungsi

Parameter mewakili variable yang nilainya akan diolah. untuk menggunakan parameter contohnya adalah seperti ini :

```
<?php
function testing($nilai)
{
    echo "Tinggi badan anda adalah $nilai";
}
$tinggibadan = 170 ;
testing($tinggibadan);
?>
```

kita lihat hasilnya di browser mozilla firefox dengan parameter dalam fungsi anda dapat mengolah apa saja, dan coding jauh lebih cepat ketimbang memanggil ulang perintah-perintah nya tanpa menggunakan fungsi. Contoh lain menggunakan parameter :

```
<?php
function penjumlahan($nilai1, $nilai2)
{
    $total = $nilai1+$nilai2;
    echo "$nilai1 + $nilai2 = ".$total;
}

penjumlahan(30,20);
?>
```

dan hasilnya di browser mozilla firefox :

```
30 + 20 = 50
```

Fungsi benar-benar bermanfaat dalam pengolahan data inputan user ataupun dari database.

Mengembalikan Nilai Fungsi

Mengembalikan nilai fungsi dilakukan jika anda menginginkan nilai dari fungsi tersebut di olah dalam suatu variable tertentu. Nah biasanya nilai dalam variable nantinya akan digunakan oleh fungsi yang lain. Untuk mengembalikan nilai fungsi, digunakan suatu fungsi return. Contoh implementasinya :

```
<?php
function penjumlahan($nilai1, $nilai2)
{
    $total = $nilai1+$nilai2;
    return $total;
}

$jumlah20plus30 = penjumlahan(30,20);

echo $jumlah20plus30;
?>
```

kita lihat hasilnya di browser,

```
50
```

kemudian kita balik ke codenya.

Fungsi penjumlahan tersebut akan mengembalikan nilai dari penjumlahan antara **\$nilai1** dan **\$nilai2**. Sehingga nilai dari fungsi dapat diimplementasikan sebagai nilai dari suatu variable **\$jumlah20plus30**. Dan hasil akhirnya adalah nilai si fungsi di ambil dari nilai return fungsi tersebut. Sehingga nilai dari fungsi tersebut dapat dimanfaatkan untuk suatu kebutuhan lainnya.

2.10 Mengenal array

Jika variable itu cuma memiliki satu nilai, maka array memiliki banyak nilai dari elemen. Array, merupakan suatu kumpulan data variable yang tersusun secara index. nanti akan saya jelaskan maksudnya itu apa. Array dapat menampung variable yang tipe datanya berbeda-beda.

2.10.1 Syntax array

Untuk membuat array, berikut adalah syntaxnya :

```
array(elemen)
```

contohnya adalah seperti ini :

```
$tahun = array(1924, 2010, 2012, 2024, 2050);
```

saya akan jelaskan, array di mulai dari index urutan 0.

```
array(1924, 2010, 2012, 2024, 2050);
```

sehingga :

```
index 0 --> di miliki oleh array bernilai 1924  
index 1 --> di miliki oleh array bernilai 2010  
index 2 --> di miliki oleh array bernilai 2012  
index 3 --> di miliki oleh array bernilai 2024  
index 4 --> di miliki oleh array bernilai 2050
```

Sehingga apabila anda ingin mengambil nilai dari array, misalkan 2024, anda cukup dengan menampilkan nama variable nya kemudian di tambahkan index dibelakang variable tersebut, seperti ini :

```
$tahun[3]; --> mewakili 2024.
```

Sehingga apa bila di aplikasikan dalam code php:

```
<?php  
$tahun = array(1924, 2010, 2012, 2024, 2050);  
echo $tahun[3];  
?>
```

Sehingga hasilnya nanti di browser adalah :

```
2024
```

selain angka, php array pun dapat memuat berbagai variable dengan tipe data yang lain.
Contoh lainnya :

```
<?php
$mahasiswa = array("Loka Dwiartara", "Laki-
laki", "24/01/1987", "B", 3.41, "Bot Technology");
for ($x=0;$x<6;$x++)
{
    echo $mahasiswa[$x]. "<br>";
}
?>
```

hasilnya adalah :

```
Loka Dwiartara
Laki-laki
24/01/1987
B
3.41
Bot Technology
```

Pengulangan tadi sebetulnya hanya mengambil isi dari array satu demi satu.
Sehingga yang terjadi sebetulnya adalah sama seperti ini :

```
<?php
$mahasiswa = array("Loka Dwiartara", "Laki-
laki", "24/01/1987", "B", 3.41, "Bot Technology");
echo $mahasiswa[0]. "<br>";
echo $mahasiswa[1]. "<br>";
echo $mahasiswa[2]. "<br>";
echo $mahasiswa[3]. "<br>";
echo $mahasiswa[4]. "<br>";
echo $mahasiswa[5]. "<br>";
?>
```

Dan kita lihat hasilnya, sama dengan script yang menggunakan pengulangan bentuk for ada beberapa syntax dalam php, yang biasa digunakan untuk mengambil isi dari array, selain dengan pengambilan langsung seperti cara tadim ataupun menggunakan looping seperti yang saya jelaskan barusan. Kita dapat menggunakan fungsi dalam pengolahan array, dengan cara memanfaatkan key dari tiap-tiap array, dan hal ini membutuhkan satu

pengulangan tersendiri.

```
<?php
$mahasiswa = array( "Loka Dwijartara", "Laki-
laki", "24/01/1987", "B", 3.41, "Bot Technology" );
for($x=0;$x<6;$x++)
{
    echo "Index array mahasiswa ke : ".key($mahasiswa)." Berisi
    ".current($mahasiswa)."<br>";
    next($mahasiswa);
}
?>
```

kita lihat hasilnya di browser, ini merupakan pengambilan array dari index terendah yakni 0, ke paling akhir yakni 5. kita akan coba membaliknya, jika tadi mulai dari 0 - 5, kita mulai dari 5 dan mundur hingga urutan ke 0, dengan memanfaatkan fungsi end(), dan prev();

Kita coba implmentasikan :

```
<?php
$mahasiswa = array( "Loka Dwijartara", "Laki-
laki", "24/01/1987", "B", 3.41, "Bot Technology" );
end($mahasiswa);

for($x=0;$x<6;$x++)
{
    echo "Index array mahasiswa ke : ".key($mahasiswa)." berisi
    ".current($mahasiswa)."<br>";
    prev($mahasiswa);
}
?>
```

dengan end(), maka array akan dibawa ke paling akhir, kemudian di bawa mundur ke index terendah dengan fungsi prev(). Selain fungsi end(), prev(), dan next(), kita juga dapat menggunakan fungsi count() yang digunakan untuk menghitung jumlah variable yang ada dalam array, sebagai contohnya, kita akan menghitung jumlah array yang berada pada array mahasiswa :

```
<?php
$mahasiswa = array( "Loka Dwijartara", "Laki-
laki", "24/01/1987", "B", 3.41, "Bot Technology" );
$jumlaharray = count($mahasiswa);
echo $jumlaharray;
```

```
?>
```

Mengakses Array dalam variable yang terpisah

Ada sedikit pertanyaan mungkin, masih seputar array, mungkin anda menginginkan array tersebut di akses kedalam variable variable yang terpisah, anda dapat menggunakan fungsi list(); implementasinya seperti ini :

```
<?php
$mahasiswa = array("Loka Dwiartara", "Laki-laki", "24/01/1987", "B", 3.41, "Bot Technology");

list($nama, $jeniskelamin, $tanggallahir, $poin, $IP, $spesialisasi) = $mahasiswa;

echo $nama."<br>";
echo $jeniskelamin."<br>";
echo $tanggallahir."<br>";
echo $poin."<br>";
echo $IP."<br>";
echo $spesialisasi."<br>";
?>
```

2.10.2 Mengurutkan array

Sedikit lagi dengan array, bagaimana mengurutkan array ? hal ini digunakan ketika kita memiliki banyak data tapi data tersebut tidak berurutan, misalkan 1,3,2,4,7,8,6,5,9,10, kita ingin mengurutkan array tersebut dari terkecil hingga terbesar. solusinya adalah menggunakan fungsi sort() :

```
<?php
$data = array(1,3,2,4,7,8,6,5,9,10);
sort($data);

for($x=0;$x<=10;$x++)
{
    echo current($data)."<br>";
    next($data);
}
?>
```

2.10.3 Menggunakan foreach

Selain menggunakan looping dengan bentuk for tadi, kita pun dapat menggunakan pengulangan yang memang dikhkususkan untuk array, yakni foreach, dan biasanya foreach juga digunakan untuk menampilkan isi dari database. penggunaan foreach sangatlah mudah.

```
<?php  
$mahasiswa = array( "Loka Dwiartara", "Laki-  
laki", "24/01/1987", "B", 3.41, "Bot Technology" );  
  
foreach ($mahasiswa as $datamahasiswa)  
{  
    echo $datamahasiswa."<br>";  
}  
?>
```

mudah ya ??

2.11 Manipulasi String

Manipulasi string digunakan untuk mengolah suatu inputan yang dimasukkan oleh user atau di ambil dari database. Biasanya manipulasi ini digunakan untuk satu alasan tertentu, seperti misalkan mencocokan password, validasi kecocokan email, menampilkan text dengan format tertentu, misalkan menampilkan angka 3031924 dalam rupiah menjadi Rp 3.031.924,- dan masih banyak contoh yang lainnya.

Menghitung Jumlah karakter dalam string

Untuk menghitung jumlah karakter dalam string, digunakan fungsi `strlen()`; implementasinya adalah seperti ini :

```
<?php
$nama = "Loka Dwiartara";
$jumlahkarakter = strlen($nama);

echo "variable \$nama berisi $jumlahkarakter karakter";
?>
```

Membandingkan 2 String

Membandingkan 2 string biasanya digunakan untuk pencocokan password, atau pencocokan email. Di gunakan fungsi `strcmp()`; implementasinya seperti ini :

```
<?php
$passworddasli = "b3laj@rpHP";
$passwordinput = "belajarPHP";

$cocok = strcmp($passworddasli, $passwordinput);
if ($cocok != 1)
{
    echo "Password anda salah!";
}
else
{
    echo "Password anda benar.";
}
?>
```

ini merupakan simulasi ketika password yang berada dalam database yakni `$passowrdasli`, dicocokkan dengan cara penginputan yang dilakukan oleh user yakni `$passwordinput`.

strcmp itu membandingkan antara password database dengan password yang input, ketika password yang di input itu tidak cocok dengan yang ada di database maka yang terjadi kemudian adalah pernyataan "Password anda salah".

Mengkapitalkan string

Ini adalah istilah lain dari mengubah text yang tadinya huruf kecil semua menjadi huruf besar semua. Cara penggunaannya cukup mudah, cukup dengan menggunakan strtoupper(); Implementasinya adalah sebagai berikut :

```
<?php  
$intro = "jangan berteriak!!!!";  
$introhurufbesar = strtoupper($intro);  
echo $introhurufbesar;  
?>
```

Hasil di browser :

```
JANGAN BERTERIAK!!!!
```

cukup mudah bukan ?

Menjadikan string huruf kecil semua

Kebalikan dari strtoupper adalah strtrtolower ;
implementasinya adalah sebagai berikut :

```
<?php  
$intro = "SEDANG BERBISIK";  
$introhurufkecil = strtolower($intro);  
echo $introhurufkecil;  
?>
```

Mengubah baris baru menjadi

Ketika anda mengetikkan sebuah kalimat dalam PHP seperti ini, dengan banyak baris di dalamnya ...

```
Pada suatu hari  
saya jalan sendirian  
di tengah jalan bertemu kucing herder  
saya di kejar sampai nyusruk
```

namun ketika di tampilkan dengan menggunakan echo, seperti ini

```
<?php  
$cerita = "Pada suatu hari  
saya jalan sendirian  
di tengah jalan bertemu kucing herder  
saya di kejar sampai nyusruk";  
  
echo $cerita;  
?>
```

yang terjadi malah, semuanya di tampilkan dalam satu baris :

Pada suatu hari saya jalan sendirian di tengah jalan bertemu kucing herder saya di kejar sampai nyusruk

php memiliki fungsi nl2br(), yang menjadikan baris-baris baru menjadi
, dan

Kalimat-kalimat yang diketik ditampilkan seperti ketika di ketik. Implementasinya seperti ini

```
<?php  
$cerita = "Pada suatu hari  
saya jalan sendirian  
di tengah jalan bertemu kucing herder  
saya di kejar sampai nyusruk";  
  
echo nl2br($cerita);  
?>
```

hasilnya adalah :

Pada suatu hari
saya jalan sendirian
di tengah jalan bertemu kucing herder
saya di kejar sampai nyusruk

Memfilter tag html

Memfilter tag html biasanya digunakan ketika seorang web developer tidak menginginkan spammer merusak tampilan guestbook, dengan linknya, atau menonaktifkan tag html yang tidak dibutuhkan.

Misalkan ketika seorang spammer yang menaruh link situsnya kedalam guestbook, maka untuk menonaktifkan link tersebut cukup gunakan strip_tags(); implementasinya adalah seperti ini :

```
<?php
$situsspammer = "Silahkan <u>berkunjung</u> ke <a href=
'&nbsp;http://www.tokosandaljepit.com'> Toko Online Sandal Jepit
<strong>Aneka Warna</strong></a>" ;

$filterlink = strip_tags($situsspammer, "<strong><u>" );
echo $filterlink;
?>
```

Dan silahkan save lalu lihat hasilnya di browser anda. **strip_tags()** memfilter hanya tag yang didefinisikan saja yang ditampilkan.

Format penggunaan **strip_tags** adalah sebagai berikut :

```
strip_tags(kalimat yang akan di filter, tag yang boleh ditampilkan)
```

Sehingga pada contoh kasus ini, diluar dari tag **** dan **<u>** maka tag tersebut tidak ditampilkan.

Memecah string ke dalam array dengan explode()

Bagaimana cara untuk memecah string, menjadi kumpulan elemen dalam array? Dengan cara menggunakan **explode()**. String yang berisi kata-kata akan dipecah, sehingga menjadi array-array, yang nantinya digunakan untuk kebutuhan tertentu.

Sebagai contoh :

```
<?php
$belajar = "saya sedang belajar bahasa pemrograman PHP";
$arraybelajar = explode(" ", $belajar);
echo $arraybelajar[5];
?>
```

sedikit saya jelaskan, **explode()** menjadikan spasi sebagai pemisah antara elemen array yang satu dengan lainnya, sehingga apabila terdapat spasi maka kata-kata tersebut akan di pecah sehingga menjadi array. Contoh lainnya

```
<?php
$tanggal = "03/Maret/1924";
$tanggalarray = explode("/", $tanggal)
echo $tanggalarray[2];
?>
```

Menggabungkan array menjadi satu kesatuan dengan implode()

Jika explode bertugas untuk memecah menjadi elemen array, implode berfungsi sebaliknya, yakni menggabungkan array menjadi satu string, contohnya adalah seperti ini:

```
<?php
$arraybelajar = array("saya", "sedang", "belajar", "bahasa",
"pemrograman", "PHP");
$kalimatbelajar = implode(" ", $arraybelajar);
echo $kalimatbelajar;
?>
```

simple and easy. right? Contoh yang lain :

```
<?php
$arraytanggal = array(03, "Maret", 1924);
$tanggal = implode("-", $arraytanggal);
echo $tanggal;
?>
```

Array yang berbeda tipe datanya dapat di satukan menjadi satu kesatuan string dengan menggunakan implode(). Mudah kan?

Mencari Posisi Suatu Karakter Dengan Strpos()

Untuk mencari posisi suatu karakter atau string anda dapat menggunakan strpos(). Contohnya adalah sebagai berikut :

Misalkan anda ingin mencari posisi kata PHP, dari kalimat saya sedang belajar PHP.

```
<?php
$kalimat = "Saya sedang belajar PHP";
$katayangdicari = "PHP";
$posisi = strpos($kalimat, $katayangdicari);
echo $posisi;
?>
```

Menampilkan sebagian isi string dengan substr()

Anda dapat menampilkan sebagian isi dari suatu string, mungkin bagi anda yang pernah membuka suatu situs biasanya di awal situs ditampilkan intro dari suatu artikel,

dengan keterangan read more, atau baca selengkapnya, hal ini menggunakan substr(), atau untuk memfilter nomor hp, dengan cara tidak menampilkan keseluruhan nomornya, kemudian diganti dengan karakter xxx, dan lain sebagainya.

Cara penulisannya adalah seperti ini.

```
substr(kalimat, nilai awal, nilai akhir atau range )
```

contohnya :

```
<?php  
$kalimat = "saya sedang mengetik menggunakan keyboard, guna mempelajari  
bahasa PHP";  
$potongkalimat = substr($kalimat, -3);  
echo $potongkalimat ;  
?>
```

substr(\$kalimat, -3) di sini berarti, tampilkan sebagian isi variable kalimat yakni 3 karakter dari belakang. Dan hasilnya adalah seperti ini ...

```
PHP
```

contoh yang lain :

```
<?php  
$kalimat = "saya sedang mengetik menggunakan keyboard, guna mempelajari  
bahasa PHP";  
$potongkalimat = substr($kalimat, 0, 41);  
echo $potongkalimat ;  
?>
```

kita lihat hasilnya di browser adalah seperti ini ...

```
saya sedang mengetik menggunakan keyboard
```

Ini berarti script ini memiliki fungsi untuk menampilkan mulai dari karater dari posisi 0 dengan range 40 karakter didepannya, contoh yang lain :

```
<?php  
$kalimat = "saya sedang mengetik menggunakan keyboard, guna mempelajari  
bahasa PHP";  
$potongkalimat = substr($kalimat, 5, 6);  
echo $potongkalimat ;  
?>
```

Lihat hasilnya di browser, script ini berfungsi menampilkan sebagian isi dari variable \$kalimat, substr(\$kalimat, 5, 6); berarti, menampilkan mulai dari posisi 5, dan rangenya 6 karakter, dimulai dari posisi 5 itu sendiri. Sehingga pada posisi 5 ada huruf s, dan rangenya 6 karakter adalah s,e,d,a,n,g sehingga yang diambil adalah kata sedang.

Menimpa karakter dengan karakter yang lain dengan str_replace()

Bagaimana jika anda ingin melakukan manipulasi dari suatu string. Misalkan ingin menimpa suatu kata dengan kata yang lain, misalkan mengubah kata bermain dalam kalimat "saya sedang bermain" menjadi "saya sedang belajar". Anda dapat memanfaatkan str_replace(); contoh penggunaanya adalah seperti ini

```
str_replace("kata/karakter asli", "kata/karakter yang baru",
"kalimat");
```

implementasinya

```
<?php
$kalimat = "saya sedang bermain";
$kalimatbaru = str_replace("bermain", "belajar", $kalimat);
echo $kalimatbaru ;
?>
```

dan hasilnya adalah :

```
saya sedang belajar
```

Manipulasi string dengan trim()

trim() digunakan untuk menghilangkan spasi, dikiri atau dikanan, biasanya digunakan untuk proses pengolahan inputan username dan password yang dilakukan oleh user, biasanya ini terjadi ketika user tidak sengaja/lupa menekan tombol spasi di keyboard, untuk menanganinya anda dapat menggunakan trim()
contoh penggunaanya :

```
<?php
$password = " iniadalahpassword ";
$passworddasli = trim($password);
echo $passworddasli;
?>
```

hasil dari script di atas adalah spasi akan hilang. Anda pun dapat menghilangkan spasi di

kiri dengan ltrim(), dan dikanan dengan rtrim();

2.12 Syntax Alternatif untuk PHP

2.12.1 Alternatif echo

Kita biasa menggunakan echo untuk proses menampilkan isi dari suatu variable, sebetulnya ada alternatif dalam php untuk menampilkan isi dari suatu variable, anda dapat menggunakan print, namun proses print sedikit lebih lama dari echo. Meskipun tidak terasa karena kecepatan komputer yang sudah sedemikian canggih, tapi tetap proses print memakan waktu 15% sedikit lebih lama dari echo.

Contoh penggunaan print sama dengan echo :

```
<?php
print "menampilkan kalimat";
?>
```

Namun para web developer cenderung menggunakan echo ketimbang print. Bukan karena kecepatan. Tapi karena web developer lebih malas mengetik 5 huruf ketimbang 4 huruf. :d aneh ya? Tapi memang itu faktanya.

2.12.2 Double Quote dan One Quote

Apa perbedaan antara double quote yang diwakili dengan tanda " dengan one quote yang di wakili oleh tanda ' One quote hanya dapat menampilkan karakter aslinya, bukan isi dari variable. Kita buktikan :

```
<?php
$nilai = 90;
echo '$nilai';
?>
```

Apabila kita lihat dibrowser, one quote, atau kutip satu itu tidak menampilkan isi dari \$nilai, melainkan karakter aslinya. Namun dengan double quote anda dapat menampilkan isi variablenya :

```
<?php
$nilai = 90;
echo "$nilai";
?>
```

mungkin kedepannya anda akan memproses kutip dua didalam kutip dua, seperti menyisipkan html di dalam echo. seperti ini misalnya :

```
<?php  
echo "<a href=\"http://www.ilmuwebsite.com\">Ilmuwebsite.com</a>" ;  
?>
```

mari kita lihat di browser, hasilnya adalah error. bagaimana agar kita dapat menggunakan tanda kutip dua, didalam kutip dua? Solusinya adalah menggunakan spesial karakter '\'. Kita buktikan :

```
<?php  
echo "<a href=\"\http://www.ilmuwebsite.com\\\">Ilmuwebsite.com</a>" ;  
?>
```

2.13 Bekerja dengan File

Di sub bab ini akan saya jelaskan bagaimana kita bekerja dengan file. mulai dari membuka file, menutup file mengecek apakah file tersebut ada, kemudian menulis isi file, ataupun menghapus file. itu semua akan saya jelaskan di sini.

Mengecek File

Untuk mengecek file, ada atau tidaknya file tersebut dalam direktori/path yang ditentukan, biasanya seorang web developer cukup menggunakan fungsi `file_exists()`, untuk mengecek apakah file tersebut ada.

Siapkan terlebih dahulu sebuah file kosong di **c:\xampp\htdocs\latihanphp** berikan nama **tes.txt**, kemudian selanjutnya silahkan buat script php nya seperti ini :

```
<?php
    if (file_exists("tes.txt") == 1)
    {
        echo "File-nya ada...";
    }
    else
    {
        echo "Filennya tidak ada ...";
    }
?>
```

`file_exists("tes.txt")` akan mengecek apakah file nya ada atau tidak, jika bernilai 1, maka akan di tampilkan "**File-nya ada...**"; jika bernilai 0, maka akan ditampilkan filennya tidak ada. Atau bisa juga di tulis seperti ini, tidak menggunakan operator perbandingan `==`.

```
<?php
    if (file_exists("tes.txt"))
    {
        echo "File-nya ada...";
    }
    else
    {
        echo "Filennya tidak ada ...";
    }
?>
```

Apakah File atau Direktori??

Untuk mengecek apakah file atau direktori terhadap nama file atau direktori kita dapat menggunakan `is_file` untuk mengecek file atau bukan dan `is_dir` untuk mengecek direktori atau bukan. Implementasinya masih menggunakan file yang tadi yakni `tes.txt`:

```
<?php
if (is_file("tes.txt"))
{
    echo "Ini adalah file!";
}
?>
```

Untuk mengecek direktori atau bukan, silahkan anda buat sebuah direktori dengan nama **direktorites** di dalam **c:\xampp\htdocs\latihanphp**.

Kemudian untuk mengecek direktorites itu adalah direktori atau bukan anda dapat mengetikan script memanfaatkan **fungsi is_dir()**

```
<?php
if (is_dir("direktorites"))
{
    echo "Ini adalah direktori!";
}
?>
```

Membuat dan menghapus file

Untuk membuat file, anda dapat menggunakan fungsi `touch()`, dan untuk menghapus file yang ada, anda dapat menggunakan fungsi `unlink()`; implementasinya adalah seperti ini, script untuk membuat file :

```
<?php
touch("filecontoh.txt");
if (file_exists("filecontoh.txt"))
{
    echo "File sudah ada!";
}
?>
```

Lihat hasilnya di browser anda. Untuk menghapus file yang sudah ada, anda dapat menggunakan unlink().

Implementasinya, kita menggunakan file yang telah kita buat sebelumnya, yakni mencoba untuk menghapus **filecontoh.txt**

```
<?php
if (file_exists("filecontoh.txt"))
{
    echo "Filennya ada!<br/>";
}

// menghapus file
unlink("filecontoh.txt");

// mengecek kembali keberadaan filenya
if (file_exists("filecontoh.txt"))
{
    echo "File sudah ada!";
}
else
{
    echo "File sudah tidak ada ... ";
}
?>
```

Membuka File

Ada beberapa fungsi untuk membuka file, mulai dari membuka file saja, kemudian ada juga fungsi yang dapat membuka kemudian lalu menulis isi file, ada juga fungsi untuk membuka file sambil menulis, dan lain-lain.

Membuka file saja anda dapat menggunakan syntax :

```
$fp = fopen( NAMA_FILENYA, 'r' );
```

dengan fungsi fopen(), dan parameter r di dalamnya, r disini hanya hanya dapat membuka file saja dan meletakkan pointer atau kurSOR di awal file.

Membuka/membaca file kemudian menulis file tersebut anda dapat menggunakan syntax:

```
$fp = fopen( NAMA_FILENYA, 'r+' );
```

parameter r+ ini memungkinkan kita membaca file kemudian menulisi file tersebut dengan meletakkan kursor atau pointer pada awal file.

Menulis file saja anda dapat menggunakan syntax :

```
$fp = fopen( NAMA_FILENYA, 'w' );
```

parameter w memungkinkan anda dapat menulis file, dengan meletakkan kursor di awal file.

Membaca dan menulis file sekaligus anda dapat menggunakan syntax seperti ini :

```
$fp = fopen( NAMA_FILENYA, 'w+' );
```

parameter w+ memungkinkan anda dapat membaca dan menulis file, dengan meletakkan kursor di awal file.

Menulis file saja anda dapat menggunakan syntax :

```
$fp = fopen( NAMA_FILENYA, 'a' );
```

parameter a memungkinkan anda dapat menulis file, namun meletakkan kursor di akhir file.

Membaca dan menulis file sekaligus anda dapat menggunakan syntax seperti ini :

```
$fp = fopen( NAMA_FILENYA, 'a+' );
```

parameter w+ memungkinkan anda dapat membaca dan menulis file, namun php meletakkan kursornya di akhir file.

Membaca File dengan feof() dan fgets()

Oke kita akan coba langsung mengimplementasikan fopen(), feof(), dan fgets() secara bersamaan untuk membaca isi suatu file.

Implementasinya adalah seperti ini:

```
<?php
$file = "tes.txt";
$fileopen = fopen($file, 'r') or die ("Nggak bisa di baca");
while (!feof($fileopen))
{
    $bariskalimat = fgets($fileopen, 1024);
```

```

        echo $bariskalimat . "<br>" ;
    }
?>

```

hal yang pertama di lakukan adalah membuka file tersebut dengan **fopen(\$file, 'r')**, **r** itu parameter untuk membaca saja dan meletakkan kursor atau pointer di awal dari file. Jika file tidak bisa dibaca akan muncul error dengan peringatan "Nggak bisa dibaca".

while (!feof(\$fileopen)) memiliki fungsi untuk mengecek apakah pointer atau kurornya berada pada akhir file. parameter **r** pada **fopen** meletakkan kursor di awal, dan **while (!feof(\$fileopen))** akan terus mengecek apakah kurornya sudah sampai akhir file, jika sudah sampai akhir file maka pengulangan akan berhenti.

```
$bariskalimat = fgets($fileopen, 1024);
```

berfungsi untuk mengambil isi dari file baris demi baris menggunakan memory 1024 byte

kemudian di tampilkan dengan ;

```
echo $bariskalimat . "<br>" ;
```

ilustrasinya adalah seperti ini ... logika membuka file

paramater r, pada **fopen(\$file, 'r')** berfungsi untuk meletakkan kursor di awal dari file.

feof mengecek apakah sudah berada di akhir file ...

fgets mengambil isi file dan mengambil baris pertama ...

fopen kemudian mengcek apakah sudah berada di akhir file?

fgets mengambil isi file dan mengambil baris kedua...

fopen kemudian mengcek apakah sudah berada di akhir file?

jika **fopen** mengecek sudah berada pada baris akhir maka pengulangan berhenti.

Menulis atau menambahkan isi file dengan fputs()

Untuk menulis isi file anda dapat menggunakan **fputs()**. Untuk dapat menggunakan **fputs()** anda harus membuka filenya terlebih dahulu, dengan **fopen()**, kemudian gunakan **fputs()**, setelah itu baru tutup file dengan **fclose();**

Implementasinya adalah seperti ini :

```
<?php
```

```
$file = fopen("tes.txt", a);
fputs($file, "Welcome to the jungle");
fclose($file);

?>
```

Parameter a pada fopen("tes.txt", a); berfungsi untuk menambahkan isi file di bagian akhir. Jika anda menggunakan w atau r, maka isi file akan dihapus kemudian ditimpah dengan yang baru. Jika anda ingin menambahkannya di akhir maka gunakan a.

Mudah sekali bukan?

2.14 Bekerja dengan Date & Time

Oke, kita sudah berada dalam akhir dari bab 2, di sini saya akan menjelaskan bagaimana menggunakan Date & Time dalam PHP. PHP menyediakan fungsi-fungsi untuk menampilkan penanggalan dan waktu. Fungsi-fungsi tersebut dapat mengembalikan tanggal dan waktu sesuai dengan yang kita butuhkan.

Berikut adalah table penanggalan dan waktu disertakan dengan penjelasannya.

a	->	am atau pm
A	->	AM atau PM
d	->	Tanggal
D	->	Nama hari dalam seminggu
F	->	Nama bulan
g	->	jam 1 sampai 12 (1..12)
G	->	jam 1 sampai 23 (0..23)
h	->	jam 1 sampai 12 (01..12)
H	->	jam 1 sampai 23 (00..23)
i	->	menit
j	->	Tanggal
l	->	Nama Hari
m	->	Bulan (dalam angka)
M	->	Bulan (singkatan dari nama bulan)
n	->	Nama bulan (1..12)
O	->	waktu greenwicj
r	->	format RFC 822 (Sun, 25 Agustus 2002 16:01:07+0200)
s	->	detik (00..59)
T	->	Timezone (DMT)
Y	->	tahun dalam dua digit
Y	->	tahun dalam empat digit
z	->	hari ke x dari tahun

contoh implementasi dari date & time adalah seperti ini untuk mengetahui tanggal berapa sekarang :

```
<?php
echo date("d");
?>
```

untuk contoh yang lebih kompleks lagi ...

```
<?php
echo "Sekarang waktu menunjukkan pukul ", date("H:i A"),
"<br>", date("l, d F Y"),
"<br>Hari ke : ", date("z"), " dari tahun ", date("Y");
?>
```

Bab 3:

Menggunakan PHP

dengan MySQL

3.0 Mengenal MySql

Sebelum lebih jauh dengan mysql, terlebih dahulu kita harus mengetahui apa itu mysql. sebagai intermezo saya akan coba ceritakan sekelumit mengenai mysql. Berbicara mengenai mysql, tidak lepas dari berbicara mengenai sql (structured query language) yakni bahasa yang berisi perintah-perintah untuk memanipulasi database, mulai dari melakukan perintah select untuk menampilkan isi database, menginsert atau menambahkan isi kedalam database, mendelete atau menghapus isi database dan mengedit database.

Beberapa sistem database yang menggunakan sql diantaranya ada Ms. Access, MsSQL Server, Oracle, PostgresSQL, DB2, MySQL, Interbase dan lain sebagainya ...

Untuk menjadi seorang web developer yang handal, anda harus menguasai perintah-perintah sql. Meskipun tidak mutlak, tapi saya sangat mengajurkan anda untuk menguasai perintah mysql untuk memudahkan dalam developer aplikasi nantinya. Secara garis besar berbagai sistem database menggunakan bahasa yang sama untuk memanipulasi isi dalam database. Sama-sama menggunakan bahasa sql.

Kenapa memilih mysql??

MySQL itu merupakan salah satu sistem database yang menggunakan sql. MySQL pun dapat digunakan secara langsung dengan mengetikkan perintah atau syntaxnya melalui console. Dan bisa juga digunakan secara embeded SQL, artinya anda dapat menggunakan perintah sql dengan menyisipkannya kedalam bahasa pemrograman tertentu, PHP misalnya. Sampai saat ini PHP dan MySQL merupakan kombinasi yang menakjubkan.

Hingga dalam kurun waktu kurang dari 20 tahun. PHP dan MySQL sudah mendominasi dunia internet khususnya web development untuk web dinamis yang powerful. Bahkan bisa dibilang, internet sebagian besar terdiri dari PHP dan MySQL. Salah satu alasan mengapa ini terjadi adalah, kedua paket tersebut bersifat open source.

Sejarah MySQL

- Berawal dari proyek yang dimulai oleh kedua orang developer, yakni Michael Widenius dan David Axmark di tahun 1994. Pada awalnya mysql Di develop menggunakan bahasa C dan C++. Lagi-lagi proyek ini didasari karena ingin membuat suatu sistem database yang murah, meskipun ketika itu ada database yang powerful yakni oracle, namun database ini bersifat komersil yang harganya super mahal , dan begitu menguasai pasar.
- Dan akhirnya MySQL dirilis pertama kali pada 23 Mei 1995. Mysql begitu diminati oleh banyak orang, karena kemudahan installasi, kemudahan pemakaian sudah seperti database komersil dengan sistem RDBMS, sistem yang memungkinkan interkoneksi antara satu tabel dengan tabel lainnya, sistem ini benar-benar effisien.
- Pada 8 Januari 1998 MySQL dirilis untuk versi Windows 95 dan NT Ini menjadikan mysql itu sebagai sistem database yang low end. artinya user biasa pun dapat menggunakan mysql ini tanpa membutuhkan server tambahan. Benar-benar suatu terobosan yang membuat mysql semakin diminati banyak orang.
- Versi 3.23 beta nya di terbitkan pada Juni 2000, dan versi stabilnya dirilis pada januari 2001.
- Versi 4.0 beta nya di terbitkan pada Agustus 2002, dan versi stabilnya dirilis pada Maret 2003. Dalam dirilis kali ini mysql sudah mengincludekan unionnya,

yang memungkin satu dengan table lain dapat saling join, dan ditampilkan dalam satu tabel.

- Versi 4.1 beta nya di terbitkan pada Juni 2004, dan versi stabilnya dirilis pada Oktober 2004.
- Dalam versi ini mysql mengincludekan R-trees and B-trees, subqueries, prepared statements.
- Versi 5.0 beta nya di terbitkan pada Maret 2005, namun versi stabilnya dirilis pada Oktober 2005 cursors, stored procedures, triggers, views, XA transactions
- Versi 5.1: currently pre-production (since November 2005) (event scheduler, partitioning, plugin API, row-based replication, server log tables). Sun Microsystems acquires MySQL AB on 26 February 2008
- Tahun 2009, 11 miliar server di internet telah menginstall mysql sebagai default sistem databasenya. MySQL kini merupakan database yang sangat powerfull. Bahkan google sudah meracik MySQL sehingga menjadi database pencarian milik google. MySQL menjadi sangat hebat. Saat ini.

3.1 Struktur dan Syntax MySQL

Struktur MySQL

MySQL Sama dengan bahasa pemrograman lainnya yang juga sama-sama memiliki tipe data.

Tipe Data String

char, memiliki panjang 255
varchar, memiliki panjang 255
tinytext, memiliki panjang 255
text, tidak terbatas
mediumtext, 1 juta
longtext, 4 Milyar

Tipe Data Numerik

int, -2 M sampai dengan 4 M
tinyint, -128 sampai dengan 255
mediumint, -8 Juta sampai dengan 8 Juta
bigint, -92 Triliyun sampai dengan 92 Triliyun
float, bilangan desimal positif, bilangan real, memiliki koma-komaan ...
double, bilangan desimal negatif dan positif, dan memiliki koma-komaan ...

Tipe Data Date dan TIme

date, hanya memuat tanggal saja, dengan formatnya seperti ini tahun-bulan-tanggal
time, hanya memuat waktu saja, formatnya HH:MM:SS (Jam:Menit:Detik)
datetime, memuat tanggal dan waktu, formatnya Tahun-Bulan-Hari Jam:Menit:Detik
year,

Tipe Key

Primary Key, Kunci yang menghubungkan antara 1 table dengan table lainnya

Auto Increment

Auto Increment menyebabkan pertambahan nilai urutan secara otomatis, suatu record memiliki nilai 1, kemudian jika ada record yang lain lagi di insertkan ke dalam database, maka urutannya akan menjadi 2 dan begitu seterus-seterusnya

Syntax MySQL

Pada bagian ini akan saya akan coba jelaskan beberapa syntax MySQL yang sering digunakan dalam membuat website, maupun aplikasi-aplikasi berbasis website, baik dalam melakukan pembuatan database, penambahanisni database, untuk menghapus isi database, ataupun untuk menghapus database itu sendiri. Syntax yang kita gunakan adalah :

- create
- use
- show
- drop
- alter
- select
- insert
- update
- delete

dibagian ini saya baru akan menjelaskan teorinya saja, di subbab selanjutnya akan saya jelaskan secara praktiknya.

Create

ini digunakan untuk membuat database maupun table. Database yang memiliki sistem RDBMS di mana telah saya terangkan di awal bab ini, bahwa RDBMS itu sistem database yang berisi informasi dalam bentuk table-table yang saling berkaitan. Jadi langkah awal dalam membuat database adalah menggunakan fungsi **create;** syntaxnya seperti ini :

```
create namadatabase;
```

untuk membuat table syntaxnya seperti ini :

```
create table(field1(ukuran), field2(ukuran), ...)
```

Use

Ini digunakan untuk menggunakan database yang telah dibuat sebelumnya, yang nantinya database ini akan di edit atau di modifikasi.

syntaxnya itu seperti ini :

```
use namadatabase;
```

Drop

Ini digunakan untuk menghapus database ataupun table yang ada. Syntaxnya adalah seperti ini :

```
drop database;
```

untuk menghapus table gunakan :

```
drop namatable;
```

Alter

Alter ini akan dijelaskan secara terperinci di subab selanjutnya yang membahas membuat database membuat table, dan mengedit keduanya. Alter berfungsi untuk memodifikasi table yang telah di buat, modifikasi nya seperti menambahkan field, mengganti size dari suatu field, menghapus field, dan mengganti nama field.

Sebagai contoh untuk mengganti size dari suatu field adalah seperti ini :

```
alter table mahasiswa change nama nama_mahasiswa varchar(75)
```

ini berarti, ganti tipe dari field nama menjadi nama_mahasiswa dengan tipe varchar yg memiliki ukuran 75

Select

Di gunakan untuk menampilkan isi dari suatu table, bisa dengan kriteria tertentu bisa juga dapat menampilkan keseluruhan tanpa adanya kriteria. Penggunaan standarnya seperti ini:

```
select * from nama_table
```

Penggunaan ketika ingin menampilkan berdasarkan kriteria adalah seperti ini :

```
select * from nama_table where field=kriteria
```

Untuk menampilkan nilai terbanyak :

```
select max(field) from tablenya
```

Untuk menjumlah isi record salah satu field dari suatu table :

```
select sum(field) from tablenya
```

untuk menghitung berapa jumlah record yang ada dari suatu table :

```
select count(*) from tablenya
```

Insert

insert ini digunakan untuk mengisi record suatu table, syntaxnya seperti ini

```
insert into nama_table(field1, field2, ...) values(nilai1, nilai2, ...)
```

contoh penggunaanya seperti ini :

```
insert into mahasiswa(nama, nilai) values('Loka Dwiartara','B');
```

ini berarti masukkan kedalam table mahasiswa, kedalam field id berisi Loka Dwiartara, dan kedalam field nilai berisi B.

Update

digunakan untuk memperbarui isi dari suatu record, syntaxnya seperti ini :

```
update nama_table set field=nilaibaru where field=kriteria
```

contoh penggunaanya adalah seperti ini :

```
update mahasiswa set nilai='A' where nama='Loka Dwiartara';
```

ini berarti perbarui atau update table mahasiswa ganti nilainya menjadi A dari mahasiswa bernama Loka Dwiartara.

Delete

digunakan untuk menghapus suatu record dari suatu table penggunaanya seperti ini :

```
delete from namatable where field=kriteria;
```

Oke, saya telah menjelaskan semua fungsi dari syntax mysql, kita akan coba praktikan semuanya.

3.2 Membuat Database, membuat table

Dalam membuat suatu aplikasi berbasis web, dibutuhkanlah database yang dapat menyimpan data-data yang nantinya diolah untuk dijadikan informasi yang bermanfaat. Sebelumnya kita telah belajar bagaimana membuat database, di sini kita akan praktikan semuanya, mulai dari membuat database.

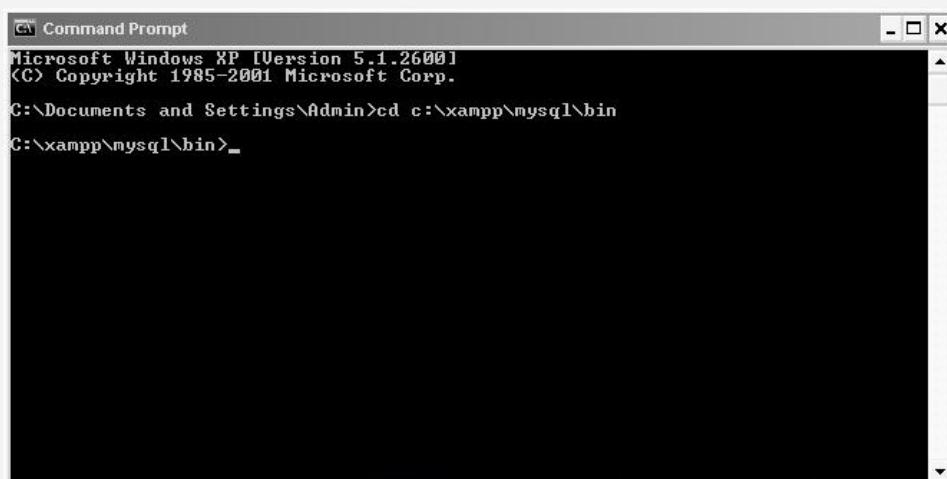
Tapi sebelum membuat database kita pastikan terlebih dahulu apakah services mysqlnya telah running melalui xampp control panel.

Setelah semuanya berjalan, kita akan coba masuk kedalam start | run | cmd. Kemudian masuk ke direktori :

```
c:\xampp\mysql\bin
```

dengan cara :

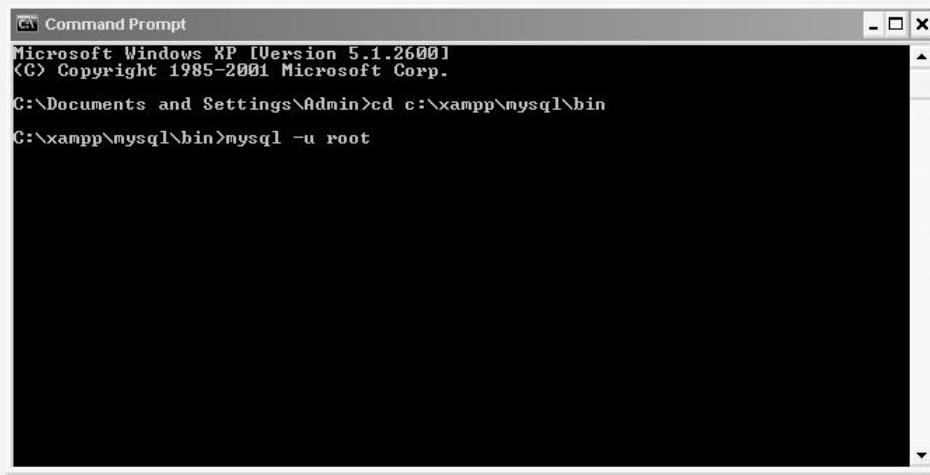
```
cd c:\xampp\mysql\bin
```



Gambar 3.1 (Dos 1)

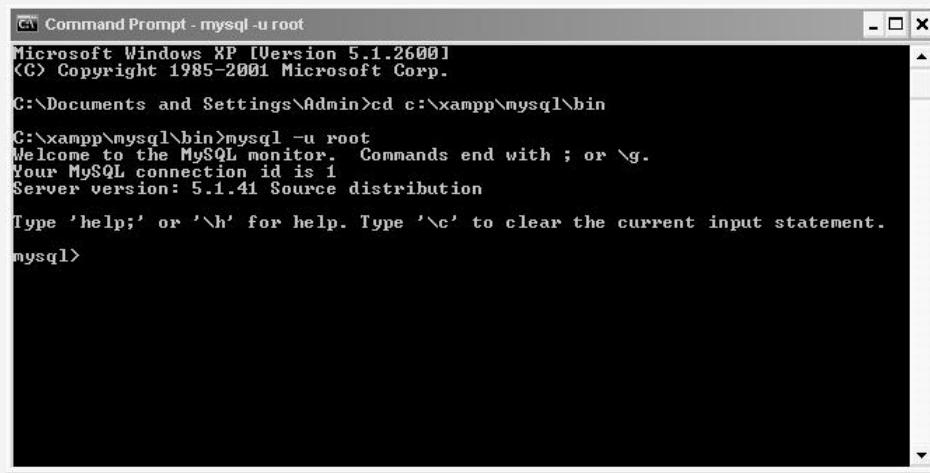
kemudian ketik :

```
mysql -u root
```



Gambar 3.2 (Dos 2)

Sehingga nanti akan tampil :



Gambar 3.2 (Dos 2)

Oke berhasil, kita akan buat satu buah database bernama mahasiswa, kenapa mahasiswa? karena nantinya database ini kita akan kembangkan menjadi suatu sistem informasi yang cukup besar, dan akan saya jelaskan bagaimana membuat suatu sistem informasi berikut membuat blog dengan data-data mahasiswa.

Oke kita akan buat database, cara nya seperti (cukup ketik yang di cetak tebal saja):

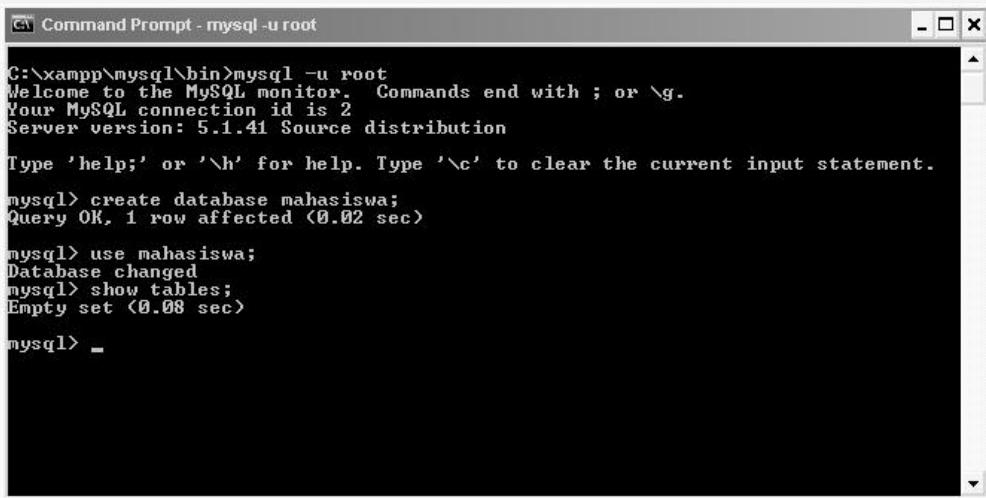
```
mysql> create database mahasiswa ;
```

Kemudian kita akan menggunakan database tersebut untuk dibuat table nya. Caranya seperti ini :

```
mysql> use mahasiswa;
```

lalu untuk melihat database tersebut sudah memiliki table apa saja, kita cukup gunakan syntaks seperti ini :

```
mysql> show tables;
```



```
C:\xampp\mysql\bin>mysql -u root
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.1.41 Source distribution

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

mysql> create database mahasiswa;
Query OK, 1 row affected (0.02 sec)

mysql> use mahasiswa;
Database changed
mysql> show tables;
Empty set (0.08 sec)

mysql> -
```

Gambar 3.3 (Dos 3)

kita buat table-nya contohnya :

strukturnya seperti ini :

nama table : tbl_mhsiswa

field	tipe	size
id_mhs	int	5
nama_mhs	varchar	50
jenis_kelamin	varchar	10
tgl_lahir	date	-
alamat	varchar	200

Kita lihat disini **id_mhs** nya adalah **int** yakni hanya untuk bilangan bulat saja, dan besarnya adalah 5, ini mengartikan bahwa maksimal jumlah mahasiswa adalah sebanyak 5 digit, yakni 99.999 record mahasiswa **id_mhs** ini berfungsi sebagai penomoran dari mahasiswa, urutan mulai dari 1 hingga sekian.

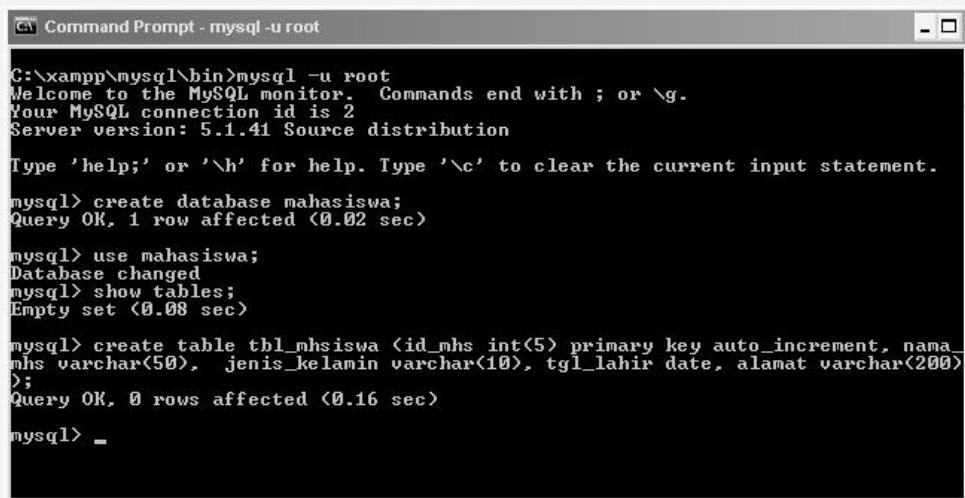
Selanjutnya ada **nama_mhs**, bertipe **varchar**, karena mahasiswa itu terdiri huruf, memiliki ukuran 50 ini digunakan sebagai perwakilan nama mahasiswa, seperti misalnya '**Loka Dwiartara**' terdapat **14 digit huruf**, dan asumsi saya adalah paling banyak itu nama orang adalah berjumlah 50. Misalnya ... "**Sukaryo Mangunkusumo SungTolodo Makan Cokor Kebo**" berjumlah 49 digit huruf, termasuk spasi didalamnya.

Begitu juga dengan **jenis_kelamin** dan alamat, namun untuk **tgl_lahir**, kita menggunakan tipe nya **date**. **date** sudah otomatis berisi 10 digit, bentuk defaultnya adalah seperti ini **0000-00-00** sehingga jika menulis **24 Januari 1987** adalah seperti ini **1987-01-24**.

Oke anda mengerti? kita lanjut, untuk merubah struktur di atas menjadi query mysql bentuknya seperti ini :

```
create table tbl_mhsiswa (id_mhs int(5) primary key auto_increment, nama_mhs
varchar(50), jenis_kelamin varchar(10), tgl_lahir date, alamat varchar(200));
```

tampilannya ...



```
C:\xampp\mysql\bin>mysql -u root
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.1.41 Source distribution

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

mysql> create database mahasiswa;
Query OK, 1 row affected <0.02 sec>

mysql> use mahasiswa;
Database changed
mysql> show tables;
Empty set <0.08 sec>

mysql> create table tbl_mhsiswa (id_mhs int<5> primary key auto_increment, nama_mhs
varchar<50>, jenis_kelamin varchar<10>, tgl_lahir date, alamat varchar<200>);
Query OK, 0 rows affected <0.16 sec>

mysql> -
```

Gambar 3.3 (Dos 3)

Kita lihat di sini, id mhs int(5) primary key auto_increment ini berarti **id_mhs** ini akan dijadikan sebagai index dari **tbl_mhsiswa**, dan **auto_increment** berfungsi membuat no

urut secara otomatis, sehingga hasil akhirnya adalah mulai dari 1, 2, 3, 4 dan seterusnya.

Untuk melihat apakah table yang sudah dibuat sudah ada kita jalankan perintah :

```
show tables;
```

```
mysql> show tables;
+-----+
| Tables_in_mahasiswa |
+-----+
| tbl_mhsiswa          |
+-----+
1 row in set (0.01 sec)
mysql>
```

Kemudian bagaimana jika kita ingin melihat isi dari table yang sudah dibuat :

```
select * from tbl_mhsiswa;
```

```
mysql> select * from tbl_mhsiswa;
Empty set (0.00 sec)
mysql>
```

setelah membuat table, kadang anda mungkin tidak sreg dengan nama tablenya, anda bisa merubah namanya dengan syntax alter ; caranya seperti ini :

```
alter table tbl_mhsiswa rename tbl_mahasiswa;
```

```
mysql> alter table tbl_mhsiswa rename tbl_mahasiswa;
Query OK, 0 rows affected (0.01 sec)
mysql>
```

kemudian lakukan show tables;

```
show tables;
```

```
mysql> show tables;
+-----+
| Tables_in_mahasiswa |
+-----+
| tbl_mahasiswa        |
+-----+
1 row in set (0.00 sec)
```

dari `tbl_mhsiswa` kemudian di rubah menjadi `tbl_mahasiswa`, kita coba kembalikan

seperti nama table yang awal :

```
alter table tbl_mahasiswa rename tbl_mhsiswa;
```

```
mysql> alter table tbl_mahasiswa rename tbl_mhsiswa;
Query OK, 0 rows affected (0.03 sec)
mysql>
```

kemudian show tables; kembali :

```
show tables;
```

```
mysql> show tables;
+-----+
| Tables_in_mahasiswa |
+-----+
| tbl_mhsiswa          |
+-----+
1 row in set (0.00 sec)

mysql>
```

misalkan kita ingin menambahkan field yang tadinya cuma **id_mhs**, **nama_mhs**, **jenis_kelamin**, **tgl_lahir**, **alamat**, sekarang kita akan menambahkan dengan satu field baru misalnya **status**, caranya adalah:

```
alter table tbl_mhsiswa add column status varchar(30);
```

```
mysql> alter table tbl_mhsiswa add column status varchar(30);
Query OK, 0 rows affected (0.13 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

ini berarti tambahkan field status dengan tipe varchar dan memiliki ukuran 30 kedalam **tbl_mhsiswa**. Ada contoh lain misalkan kita ingin merubah nama dan ukuran dari suatu field, ingin di perbesar ukurannya atau ingin di perkecil kita cukup menggunakan perintah seperti ini :

```
alter table tbl_mhsiswa add change id_mhs id_mahasiswa int(10);
```

```
mysql> alter table tbl_mhsiswa change id_mhs id_mahasiswa int(10);
```

```
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

ini berarti ganti field **id_mhs** menjadi **id_mahasiswa** bertipe **integer** dengan ukuran 10 digit.

Oke kita kembalikan ke awal :

```
alter table tbl_mhsiswa change id_mahasiswa id_mhs int(5) not null auto_increment;
```

```
mysql> alter table tbl_mhsiswa change id_mahasiswa id_mhs int(5) not null
auto_increment;
Query OK, 0 rows affected (0.08 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Bagaimana caranya jika kita hanya ingin merubah ukurannya saja ? Cara nya seperti ini:

```
alter table tbl_mhsiswa modify id_mhs int(20) not null auto_increment;
```

```
mysql> alter table tbl_mhsiswa modify id_mhs int(20) not null auto_increment;
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

memodifikasi **id_mhs** menjadi **tipe integer** dengan sizenya 20.

Dan untuk mengapus field yang ada caranya seperti ini, misalnya kita ingin mendelete field status :

```
mysql> alter table tbl_mhsiswa drop status;
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Anda bisa melihat struktur yang ada dengan :

```
mysql> desc tbl_mhsiswa;
+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+
| id_mhs     | int(20)    | NO   | PRI | NULL    | auto_increment |
| nama_mhs   | varchar(50) | YES  |     | NULL    |                |
| jenis_kelamin | varchar(10) | YES  |     | NULL    |                |
| tgl_lahir  | date       | YES  |     | NULL    |                |
| alamat     | varchar(200) | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+
5 rows in set (0.11 sec)

mysql>
```

kita coba isi terlebih dahulu dengan menggunakan perintah insert :

```
mysql> insert into tbl_mhsiswa(nama_mhs, jenis_kelamin, tgl_lahir, alamat)
values('Loka Dwiartara', 'pria', '1987-01-24', 'Jawa Barat');
Query OK, 1 row affected (0.05 sec)
mysql>
```

kemudian kita tampilkan lagi isi dari **tbl_mhsiswa** :

```
mysql> select * from tbl_mhsiswa;
+-----+-----+-----+-----+
| id_mhs | nama_mhs      | jenis_kelamin | tgl_lahir   | alamat      |
+-----+-----+-----+-----+
|      1 | Loka Dwiartara | pria          | 1987-01-24 | Jawa Barat |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

kita coba isi kembali dengan record yang lain :

```
mysql> insert into tbl_mhsiswa(nama_mhs, jenis_kelamin, tgl_lahir, alamat)
values('Anggie Jatrasmara', 'pria', '1985-10-29', 'Jawa Timur');
Query OK, 1 row affected (0.00 sec)

mysql>
```

tampilkan kembali isinya :

```
mysql> select * from tbl_mhsiswa;
+-----+-----+-----+-----+
| id_mhs | nama_mhs      | jenis_kelamin | tgl_lahir   | alamat      |
+-----+-----+-----+-----+
|      1 | Loka Dwiartara | pria          | 1987-01-24 | Jawa Barat |
|      2 | Anggie Jatrasmara | pria          | 1985-10-29 | Jawa Timur |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

bagaimana jika kita ingin menampilkan record berdasarkan kriteria, misalkan ingin melihat mahasiswa yang alamatnya di jawa barat saja :

```
select * from tbl_mhsiswa where alamat='Jawa Barat';
```

```
mysql> select * from tbl_mhsiswa where alamat='Jawa Barat';
+-----+-----+-----+-----+
| id_mhs | nama_mhs      | jenis_kelamin | tgl_lahir   | alamat      |
+-----+-----+-----+-----+
```

1	Loka Dwiartara	pria	1987-01-24	Jawa Barat
1 row in set (0.08 sec)				

mudah ya?? apalagi jika anda sering berlatih untuk mengulang perintah-perintah yang tadi. Kita coba isi kembali sebuah record kedalam tbl_mhsiswa :

```
insert into tbl_mhsiswa(nama_mhs, jenis_kelamin, tgl_lahir, alamat) values('Deny Sarwono', 'Pria', '1986-12-09','Jawa Barat');
```

```
mysql> insert into tbl_mhsiswa(nama_mhs, jenis_kelamin, tgl_lahir, alamat)
values('Deny Sarwono', 'Pria', '1986-12-09','Jawa Barat');
Query OK, 1 row affected (0.00 sec)
```

tampilkan kembali :

```
select * from tbl_mhsiswa;
```

```
mysql> select * from tbl_mhsiswa;
+-----+-----+-----+-----+
| id_mhs | nama_mhs | jenis_kelamin | tgl_lahir | alamat |
+-----+-----+-----+-----+
| 1 | Loka Dwiartara | pria | 1987-01-24 | Jawa Barat |
| 2 | Anggie Jatrasmara | pria | 1985-10-29 | Jawa Timur |
| 3 | Deny Sarwono | Pria | 1986-12-09 | Jawa Barat |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

Kita akan coba mendelete mahasiswa yang tanggal lahirnya '1986-12-09' :

```
delete from tbl_mhsiswa where tgl_lahir = '1986-12-09';
```

```
mysql> delete from tbl_mhsiswa where tgl_lahir = '1986-12-09';
Query OK, 1 row affected (0.06 sec)

mysql>
```

kemudian untuk membuktikan sudah atau belum didelete cukup lakukan perintah yang tadi, untuk menampilkan isi table:

```
select * from tbl_mhsiswa;
```

```
mysql> select * from tbl_mhsiswa;
+-----+-----+-----+-----+
| id_mhs | nama_mhs      | jenis_kelamin | tgl_lahir   | alamat      |
+-----+-----+-----+-----+
|     1  | Loka Dwiartara | pria          | 1987-01-24  | Jawa Barat |
|     2  | Anggie Jatrasmara | pria          | 1985-10-29  | Jawa Timur |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

kita juga bisa melakukan modifikasi record yang ada dengan perintah update, sebagai contohnya, kita akan coba mengganti Nama Mahasiswa dari Anggie Jatrasmara menjadi Jatrasmara :

```
update tbl_mhsiswa set nama_mhs='Jatrasmara' where alamat='Jawa Timur';
```

```
mysql> update tbl_mhsiswa set nama_mhs='Jatrasmara' where alamat='Jawa Timur';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1    Changed: 1    Warnings: 0
```

kita buktikan :

```
select * from tbl_mhsiswa;
```

```
mysql> select * from tbl_mhsiswa;
+-----+-----+-----+-----+
| id_mhs | nama_mhs      | jenis_kelamin | tgl_lahir   | alamat      |
+-----+-----+-----+-----+
|     1  | Loka Dwiartara | pria          | 1987-01-24  | Jawa Barat |
|     2  | Jatrasmara     | pria          | 1985-10-29  | Jawa Timur |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

mudah ya??? anda bisa mengulang-ngulangnya kembali.

3.3 Bagaimana Php Bekerja Dengan Mysql

Bagaimana agar php dan mysql dapat bekerja secara bersama-sama? Jawabannya ya dengan cara mengkoneksikan nya terlebih dahulu. Setelah terkoneksi selanjutnya adalah anda dapat memodifikasi database dengan menjadikan perintah-perintah atau syntax dalam mysql yang tlah di pelajari tadi sebagai query yang ada dalam php. Bingung bukan? Tapi pertanyaan-pertanyaan anda akan clear setelah. Saya jelaskan mengenai subbab selanjutnya.

3.4 Koneksi PHP dan MySQL

Untuk melakukan koneksi antara PHP dan MySQL yang perlu dilakukan adalah dengan menggunakan beberapa fungsi dalam php yang sudah disediakan untuk menangani koneksi database, tidak hanya koneksi tapi dalam php pun sudah terdapat fungsi untuk mengupdate, mendelete, menginsert dan seputar fungsi yang dapat menjalankan syntax-syntax mysql yang telah saya jelaskan pada subbab sebelumnya.

Awalnya mungkin memang agak sulit dimengerti bagi anda yang pemula dalam programming, tapi jika sering di ulang, saya jamin anda akan cepat mahir.

Untuk melakukan koneksi antara php dan mysql sebelumnya terlebih dahulu masuk ke dalam c:\xampp\htdocs\. Kemudian kita buat sebuah folder dengan nama latihan3, kemudian buat sebuah file dengan nama **koneksi.php**

```
<?php
// untuk dapat mengkoneksikan php kepada mysql
// ada 4 informasi yang anda butuhkan,
// pertama adalah host ini adalah nama suatu komputer
// username, ini merupakan username dari mysql server yang telah terinstal
// bersamaan dengan xampp
// password sama dengan username
// dan database apa yang ingin di koneksi
// defaultnya ketika anda menggunakan xampp maka informasi tersebut berisi
// hostnya adalah localhost, usernamenya adalah root, dan passwordnya itu
// kosong
// sedangkan database yang akan kita koneksi adalah database mahasiswa yang
// telah kita buat sebelumnya
// oke langsung saja kita ketikan seperti ini

// pertama adalah hostnya
$host = "localhost";

// kemudian adalah username
```

```
$username = "root";  
  
// lalu passwordnya  
$password = "";  
  
// dan selanjutnya adalah database  
$database = "mahasiswa";  
  
// kemudian untuk mengkoneksikannya caranya adalah dengan fungsi mysql_connect  
// seperti ini  
  
$koneksi = mysql_connect($host, $username, $password);  
  
// ini berarti tolong koneksi php kepada mysql dengan informasi host,  
// username, dan password  
  
// kemudian untuk memilih database yang akan di koneksi  
// pilihlah database yang ada dalam koneksi yang sedang berlangsung, jika tidak  
// bisa maka gagal koneksi  
// mudah bukan ???  
  
$pilihdatabase = mysql_select_db($database, $koneksi);  
  
if ($pilihdatabase) echo "Berhasil";  
else echo "Gagal Koneksi";  
  
// apabila kita eksekusi di browser, jika koneksi sukses maka tampilannya  
// "Berhasil"  
// namun jika terdapat komentar Gagal Koneksi, berarti anda belum berhasil  
?>
```

3.5 Database Query

Setelah berhasil melakukan koneksi, selanjutnya kita akan coba untuk menampilkan isi database menggunakan php, pada bab 2 saya telah jelaskan secara rinci bagaimana menggunakan PHP dengan benar, kita akan aplikasikan pada subbab ini.

Di sini kita akan coba bagaimana menampilkan isi database menggunakan pengulangan jenis while masuk dulu ke dalam direktori latihan3 yang tadi, kemudian buat sebuah file dengan nama **selectquery1.php**

Untuk dapat menampilkan isi database, pertama adalah lakukan dulu koneksi, baru kemudian jalankan querynya untuk menampilkan isi databasenya.

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

$tampilan = "select * from tbl_mhsiswa";

$query_tampilan = mysql_query($tampilan);

while($hasil = mysql_fetch_array($query_tampilan))
{
    echo $hasil['nama_mhs']."<br />";
}
?>
```

mudah bukan??? anda cukup mengulang-ulang cara ini hingga anda menjadi seorang yang advanced dalam menggunakan query mysql pada php.

Kita akan coba menggunakan query yang sama untuk mengambil field yang lain...

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

$tampilan = "select * from tbl_mhsiswa";

$query_tampilan = mysql_query($tampilan);

while($hasil = mysql_fetch_array($query_tampilan))
{
    echo $hasil['nama_mhs']."<br />";
    echo $hasil['jenis_kelamin']."<br />";
}
?>
```

kita juga dapat melakukan query insert dengan menggunakan php, buat sebuah file dengan nama **insertquery.php**

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

$insert = "insert into tbl_mhsiswa(nama_mhs, jenis_kelamin, tgl_lahir, alamat)
values('Deny Sarwono', 'Pria', '1986-12-09','Jawa Barat');";

$insert_query = mysql_query($insert);

if($insert_query) {echo "Berhasil di insert";}
else {echo "Gagal Insert Record";}

?>
```

selain itu kita juga bisa melakukan query delete dengan menggunakan php, buat sebuah file dengan nama **deletequery.php**

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

$delete = "delete from tbl_mhsiswa where tgl_lahir = '1986-12-09';";
$delete_query = mysql_query($delete);

if($delete_query) echo "Record Telah berhasil di hapus...";
else echo "Record Gagal untuk dihapus..";

?>
```

3.6 Menggunakan PHPMyAdmin

Selain menggunakan mysql yang ada di command prompt, kita juga bisa menggunakan phpmyadmin, sebuah aplikasi berbasis web yang dapat memanage database, untuk dapat menggunakan phpmyadmin, cukup gunakan browser internet kesukaan anda, dalam hal ini saya menggunakan browser mozilla firefox. lalu kita isi alamatnya di address bar seperti ini :

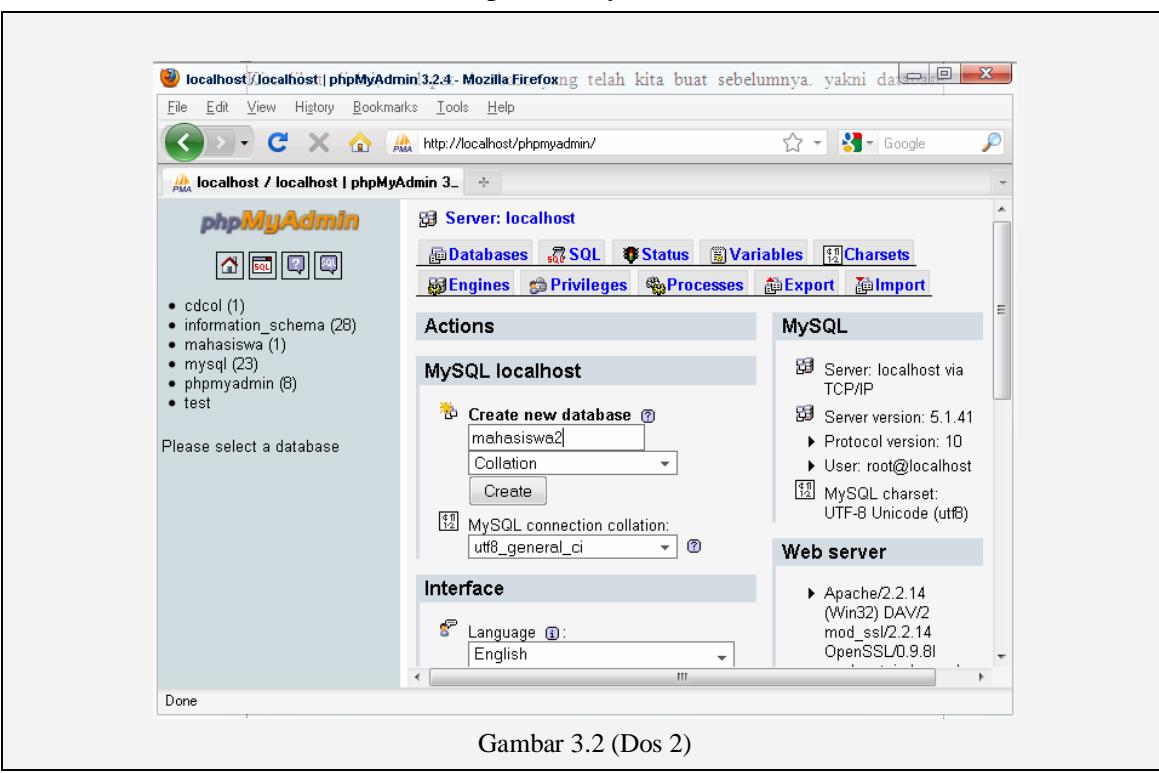
<http://localhost/phpmyadmin>

Kita lihat disini, terdapat database yang telah kita buat sebelumnya. yakni database mahasiswa. Apabila kita klik struktur di bagian sini, maka hasilnya terlihat sama dengan yang ada di command prompt mysqlnya. Sama Bukan?

Saya akan jelaskan bagaimana membuat database di phpmyadmin, bagaimana memanagenya melakukan insert record, delete record, manipulasi table dan database.

Membuat database dengan phpmyadmin

Balik ke awal halaman phpmyadmin, kemudian buat satu buah database, dengan nama mahasiswa2, caranya cukup gampang tinggal memasukkan namanya, mahasiswa2 kemudian tekan tombol **Create**, cukup mudah ya ???

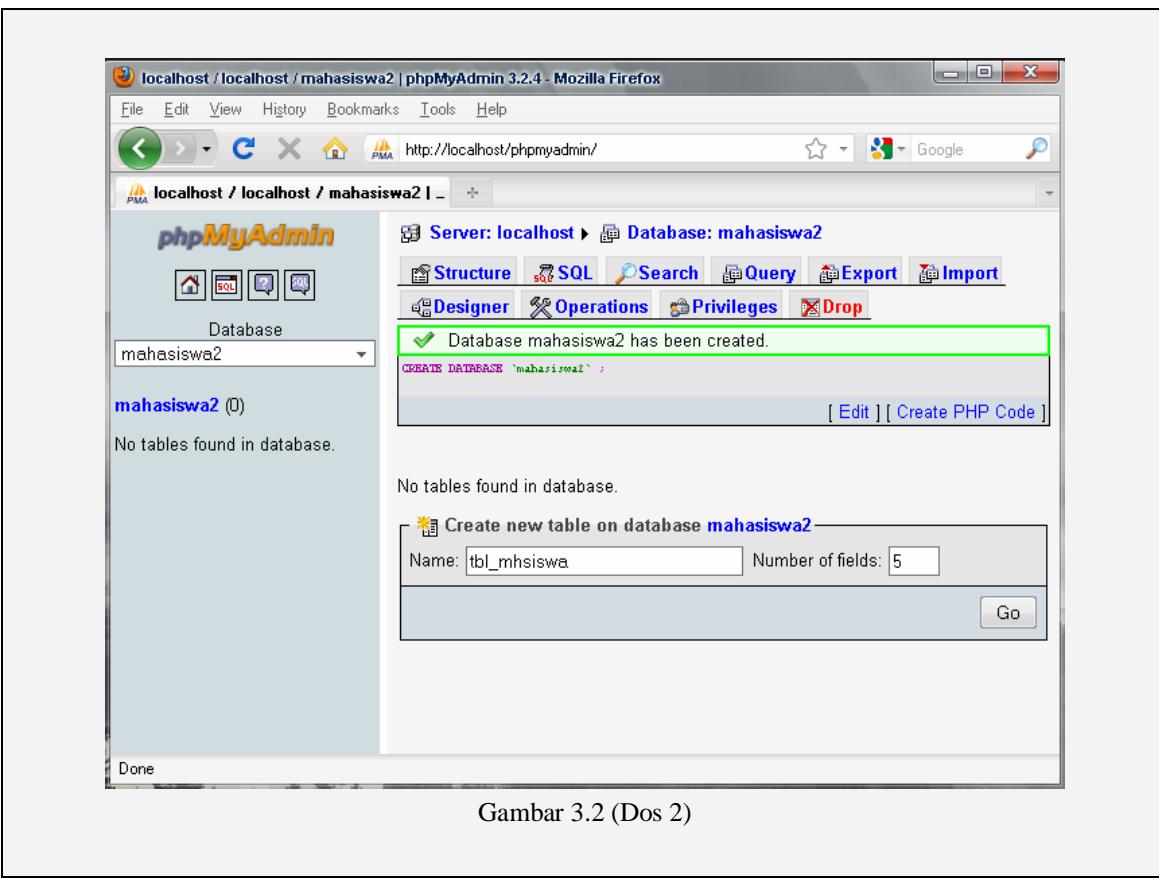


Membuat table dengan phpmyadmin

Oke dari situ kita coba langsung satu buah table yang strukturnya sama dengan tabel yang telah kita buat sebelumnya, seperti ini :

field	tipe	size
id_mhs	int	5
nama_mhs	varchar	50
jenis_kelamin	varchar	10
tgl_lahir	date	-
alamat	varchar	200

kita coba akan buat, 1 table dengan nama **tbl_mhsiswa**, dengan 5 field. Sehingga tinggal masukkan ke dalam phpmyadminnya, **nama = tbl_mhsiswa**, dan **number of fields = 5**



Gambar 3.2 (Dos 2)

Kemudian klik Go. Beres.

Bagian-bagian dari phpmyadmin

Jika dalam command prompt mysql, kita bersusah payah untuk merubah struktur suatu field menggunakan perintah-perintah, taapi dengan phpmyadmin kita cukup melakukan klak klik, pekerjaan mudah dilakukan untuk merubah struktur dari suatu table kita cukup memilih tablenya lalu pilih struktur, disini saya akan jelaskan fungsi dari masing-masing tab.

browse

berguna untuk menampilkan record yang telah di insertkan

structure

untuk melihat dan memodifikasi struktur table yang ada misalnya kita mengganti field nama_mhs, menjadi nama_mahasiswa, begitu juga apabila ingin menambah field, anda cukup mengaddnya ...

sql

ini berfungsi untuk menjalankan custom query, anda bisa menjalankan query mysql disini

insert

insert ini digunakan untuk menginsertkan record, berisi form-form yang siap anda inputkan

export

export digunakan untuk mengeksportkan database, biasanya hal ini kita temukan ketika ingin memindahkan database lokal ke database hosting, untuk dionlinekan. oke sebagai contoh seperti ini ...

import

fasilitas import ini digunakan untuk mengimportkan database yang sebelumnya telah diexportkan import berfungsi untuk merubah query-query mysql kedalam database yang real.

Bab 4:

Menampilkan Data Menggunakan Table

4.1 Mendefinisikan Tabel Heading

Dalam proses pembuatan sebuah aplikasi berbasis website, contohnya sistem informasi, seorang developer aplikasi biasanya banyak menggunakan tabel ketika akan menampilkan suatu informasi. banyak keuntungan ketika menggunakan tabel, tampilan cenderung lebih rapi, dan proses menampilkan jauh lebih mudah.

Sebelum kita memanfaatkan tabel, saya akan jelaskan dulu, bagaimana proses tabel ini digunakan. Terdapat dua bagian yang akan kita kerjakan, yakni :

- Bagian Statis
- Bagian Dinamis

No	Nama	Jenis Kelamin	Tanggal Lahir	Alamat
Contoh No	Contoh Nama	Contoh Jenis Kelamin	Contoh Tanggal Lahir	Contoh Alamat

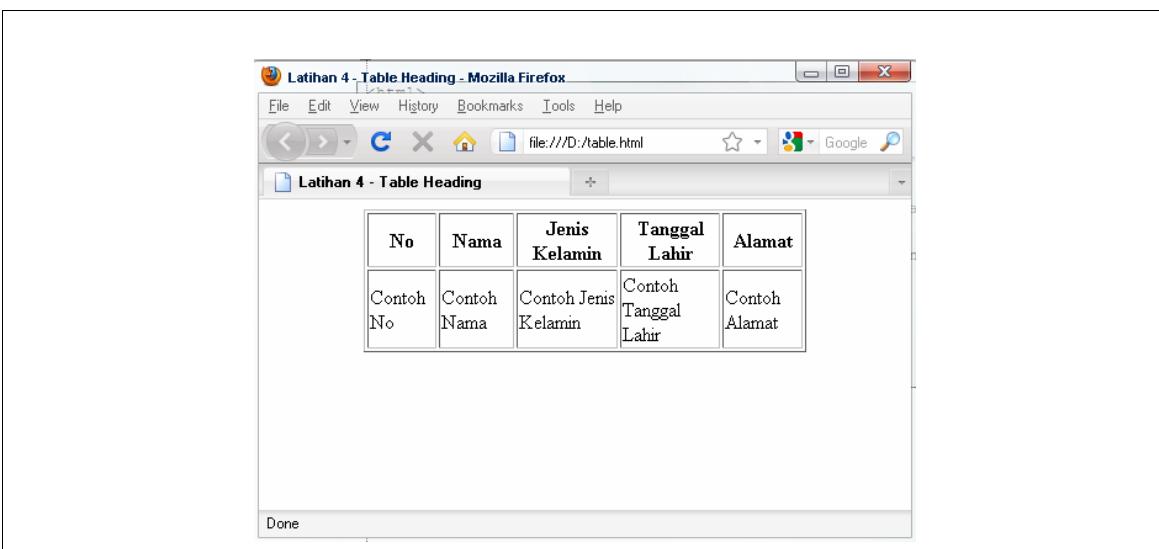
Gambar 4.1 (Table Heading)

Yang di maksud bagian statis adalah heading dari tablenya, seperti : **No, Nama, Jenis, Kelamin, Tanggal Lahir, Alamat**. Sedangkan yang dibagian bawahnya **Contoh No, Contoh Nama**, dan lain-lain itu adalah bagian yang dinamisnya. Di mana bagian yang dinamis ini adalah data yang kita ambil dari database.

Dalam mengambil data dinamis, php mengambilnya secara per-record/perbaris, sehingga untuk menampilkan datanya pun kita lakukan dengan cara perbaris, oke langsung saja kita buat table headingnya terlebih dahulu. Kemudian kita buat template/contoh data ... ini akan mempermudah kita dalam proses penampilan data nantinya, kita kerjakan bagian statis terlebih dahulu dengan membuat file tablehead.php, terlebih dahulu, di sini kita buat html dengan table didalamnya, penulis asumsikan anda telah mengetahui pembuatan table menggunakan html.

```
<html>
<head><title>Latihan 4 - Table Heading</title></head>
<body>
<table border='1' cellpadding='2' cellspacing='2' align='center'
width='70%'>
<tr><th>No</th><th>Nama</th><th>Jenis Kelamin</th><th>Tanggal
Lahir</th><th>Alamat</th></tr>
<tr><td>Contoh No</td><td>Contoh Nama</td><td>Contoh Jenis
Kelamin</td><td>Contoh Tanggal Lahir</td>
<td>Contoh Alamat</td></tr>
</table>
</body>
</html>
```

Hasilnya di browser :



Gambar 4.1 (Table Heading)

4.2 Memasukkan Data ke dalam tabel

Kita masih menggunakan isi file yang tadi, kita akan membuat bagian yang dinamis dalam menampilkan data, caranya adalah dengan mengambil isinya dari database, sebelum dapat mengambil atau menampilkan data, terlebih dahulu kita lakukan koneksi, kita akan gunakan file koneksi.php yang dibab 3 telah kita buat, kita manfaatkan file tersebut sebagai file koneksi yang akan menghubungkan php dengan mysql sehingga kita bisa menampilkan isinya kedalam table.

```
<html>
<head><title>Latihan 4 - Table Heading</title></head>
<body>
<table border='1' cellpadding='2' cellspacing='2' align='center'
width='70%'>
<tr><th>No</th><th>Nama</th><th>Jenis Kelamin</th><th>Tanggal
Lahir</th><th>Alamat</th></tr>

<!--
kita akan menampilkan isi databasenya di sini ...
dengan memanfaatkan pola contoh data yang telah kita buat sebelumnya
-->

<!-- bagian dinamis -->
<?php
// kita gunakan echo untuk dapat ditampilkan oleh php
echo "<tr align='center'><td>Contoh No</td><td>Contoh
Nama</td><td>Contoh Jenis Kelamin</td>
<td>Contoh Tanggal Lahir</td><td>Contoh Alamat</td></tr>" ;
?>
<!-- bagian dinamis -->

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

Script lengkapnya adalah sebagai berikut :

```
<html>
<head><title>Latihan 4 - Table Heading</title></head>
<body>
<table border='1' cellpadding='2' cellspacing='2' align='center'
width='70%'>
<tr><th>No</th><th>Nama</th><th>Jenis Kelamin</th><th>Tanggal
Lahir</th><th>Alamat</th></tr>

<!--
kita akan menampilkan isi databasenya di sini ...
dengan memanfaatkan pola contoh data yang telah kita buat sebelumnya
-->

<!-- bagian dinamis -->
<?php
include "koneksi.php";
$tampilan_isi = "select * from tbl_mhsiswa";
$tampilan_isi_sql = mysql_query($tampilan_isi);
while ($isi = mysql_fetch_array($tampilan_isi_sql))
{
$no = $isi['id_mhs'];
$nama = $isi['nama_mhs'];
$jeniskelamin = $isi['jenis_kelamin'];
$tanggallahir = $isi['tgl_lahir'];
$alamat = $isi['alamat'];

echo "<tr align='center'> <td>$no</td> <td>$nama</td>
<td>$jeniskelamin</td> <td>$tanggallahir</td> <td>$alamat</td> </tr>";
?>
<!-- bagian dinamis -->

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

Bab 5:

Bekerja dengan Form

5.1 Menggunakan Form

Kita berada dalam topik bahasan yang benar-benar berperan penting dalam developing website, atau developing aplikasi berbasis website. Halaman tanpa form, tidak bisa digunakan untuk menginput, jelaslah itu...

Website yang interaktif, menjadikan user dapat berperan aktif misalnya seperti mengisi poling, berkomentar pada suatu artikel, bahkan menginput berita untuk diumumkan ke orang banyak, dan contoh yang lain, dalam suatu sistem aplikasi berbasis website, form merupakan sesuatu yang layak digunakan, karena untuk melakukan suatu inputan itu tidak bisa dilakukan dengan cara yang lain.

Di sini saya akan coba jelaskan bagaimana memanfaatkan php dalam memproses form inputan dari user, pertama yang harus kita buat adalah tampilan form untuk user, kita buat satu buah file dengan nama **form1.html** kita akan buat sebuah halaman html berisi 1 buah form :

```
<html>
<head><title>Form 1 - Latihan 5</title></head>
<body>

// disini kita akan manfaatkan table untuk membuat form
// pertama buat formnya terlebih dahulu

<form method="POST" action="process1.php">
```

```

// attribut method berisi POST, ini memiliki arti, lakukan metode
// pengiriman, tapi apa yang dikirimkan itu
// tidak tampil di url browser
// saya akan jelaskan nanti di contoh yang lain, masih dalam sub bab
// ini...
// namun sementara kita diamkan seperti itu dulu, dengan method POST
// untuk action="process1.php" ini berarti kirimkan data-data yang di
// input oleh user ke file process1.php
// untuk proses selanjutnya

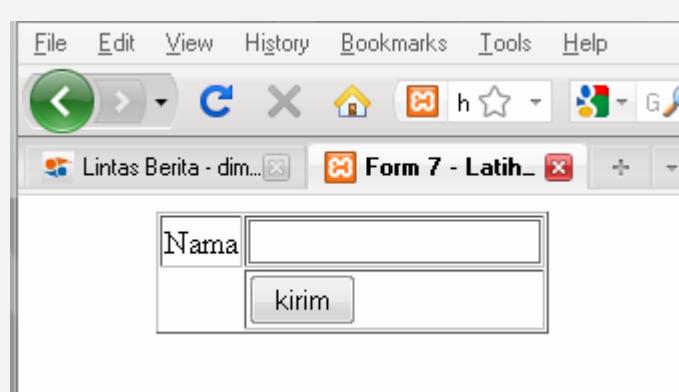
// kemudian kita buat layout form nya dengan table





```

kita lihat hasilnya di browser :



Gambar 5.1 (Form1.html)

Apabila kita isi namanya, kemudian kita klik kirim, yang terjadi adalah ... error. Ini dikarenakan kita belum membuat file **process1.php**.

Apabila kita view selection source (nama), input bertipe name itu menghasilkan textbox yang dapat menerima inputan begitu juga dengan input bertipe submit, ini berfungsi untuk mengirim data.

5.2 Memproses Form

Setelah kita membuat formnya, langkah selanjutnya adalah, buat satu buah file, yakni **process1.php**, file ini berfungsi untuk menerima inputan, lalu memprosesnya, dan menampilkan yang telah di inputkan, kemudian di dalamnya kita ketikkan ...

```
// kita awali dengan cara mendefinisikan ini adalah script php
<?php
// untuk mengambil informasi yang telah di kirimkan
// kita perlu mengetahui 2 hal,
// yang pertama, informasi yang dikirimkan itu menggunakan metode apa
// dan yang kedua, untuk menampilkan suatu data, anda perlu melihat
// attribut
// name, dari suatu input
// dalam hal ini kita ingin menampilkan data yang diinputkan melalui
// nama,
// terlebih dahulu kita
// lihat attribut name
// dari form yang telah dibuat, (lihat filenya) dan kita lihat, si
// input type
// text ini
// memiliki attribut name yang berisi "nama"
// nama inilah yang akan kita gunakan
// sehingga kita menampilkan isi yang telah di input dengan
// methodnya, kemudian isi attribut name dari input type text
// kita cukup menulisnya seperti ini ...

echo $_POST['nama'];
?>
```

kemudian kita lihat hasilnya di browser, pertama masuk kedalam formnya terlebih dahulu ... lalu kita isi namanya, dan kita kirim, yang terjadi adalah **process1.php** menampilkan inputan dari **form1.html**

5.3 Input Radio

Sebelumnya telah kita pelajari bagaimana membuat form inputan dan memproses inputan tersebut, sekarang kita akan pelajari bagaimana memproses inputan bertipe radio alias pilihan, sebetulnya tidak ada yang berbeda dengan input type text, kita cukup memanggil attribut namanya saja, oke langsung saja kita buat formnya, dengan nama **form2_radiobutton.html** terlebih dahulu :

```
<html>
<head><title>Form 2 Input RadioButton - Latihan 5</title></head>
<body>

// kita akan kirimkan dia ke process2.php, dengan method nya post,
<form action="process2.php" method="POST">
<table border='1' width='20%' cellpadding='1' cellspacing='1' align='center'>

// kemudian kita buat 2 radio button dengan nama yang sama, namun valuenya
// berbeda ...
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin"
value="laki-laki"/>laki-laki</td></tr>
<tr><td></td><td><input type="radio" name="jenis_kelamin"
value="perempuan"/>perempuan</td></tr>

// kita buat submit atau pengirimnya ...
<tr><td></td><td><input type="submit" name="kirim" value="kirim" /></td></tr>
</table>
</form>
</body>
</html>
```

kemudian kita buat satu buah file, dengan nama **process2.php**, file ini lah yang akan menampilkan jenis kelamin yang dipilih, kita cukup memakai methodnya, dan mengakses input namanya ...

```
<?php
echo $_POST['jenis_kelamin'];
?>
```

kita buka formnya :

```
http://localhost/file.latihan.php/latihanbab5/form2\_radiobutton.html
```

kemudian isi, dan kita submit, lalu lihat hasilnya di browser, ini terjadi karena yang ditampilkan itu ada value dari input type radio nya jenis_kelamin. Cukup mudah bukan???

5.4 Input Select Option

Menggunakan input select option tidak berbeda dengan radio button, namun hanya berbeda di struktur htmlnya saja. buat form htmlnya terlebih dahulu ...

```
<html>
<head><title>Form 3 Input Select Option - Latihan 5</title></head>
<body>

// kita akan mengirimkan dengan method post ke halaman process3.php
<form action="process3.php" method="POST">
<table border='1' width='20%' cellpadding = '1' cellspacing = '1'
align ='center'>
<tr><td>Jenis Kelamin</td><td>

// lalu kita membuat input select nya kita beri nama kota
<select name="kota">

// kemudian kita buat pilihannya
// option Bogor dengan valuenya bogor
<option value="bogor">Bogor</option>
// value nya inilah yang nantinya akan ditampilkan

<option value="surabaya">Surabaya</option>
<option value="yogyakarta">Yogyakarta</option>

// kita tutup dengan membuat tag select penutup
</select>
</td></tr>

// dan membuat button submitnya
<tr><td></td><td><input type="submit" name="kirim" value="kirim" />
</td></tr>
</table>
</form>
</body>
</html>
```

kita cek di browser, oke sudah mantap, selanjutnya kita buat file process3.php sekali lagi yang kita tuliskan di phpnnya adalah methodenya dan nama inputannya

```
<?php
echo $_POST['kota'];
?>
```

5.5 Input CheckBox

Input type checkbox, dapat digunakan seperti layaknya radiobutton, cukup gunakan nama untuk dapat menampilkan isinya, namun bagaimana jika kita ingin membuat banyak checklis, dan semua checklis yang terpilih itu di tampilkan ?? disinilah gunanya looping dan array. Kita buat formnya terlebih dahulu ...

```
<html>
<head><title>Form 4 Input Checkbox - Latihan 5</title></head>
<body>
<form action="process4.php" method="POST">
<table border='1' width='20%' cellpadding='1' cellspacing='1'
align='center'>

// disini ... lalu kita buat checkboxnya, dengan nama sama, namun di
// belakang namanya kita
// sisipkan dalam kurung kotak seperti ini
// sehingga nanti name ini berfungsi sebagai array

<tr><td>Status Kelulusan : </td><td><input type="checkbox"
name="lulus[]" value="SD"/>SD</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMP"/>
SMP</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMP"/>
SMP</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMA"/>
SMA</td></tr>
<tr><td></td><td><input type="submit" name="kirim" value="kirim"
/></td></tr>
</table>
</form>
</body>
</html>
```

kemudian kita buat sebuah file php dengan nama **process4.php**

```
// kita awali dengan tag php
<?php
// kemudian kita coba dengan
// menampilkan isi, dengan memanggil attribut namanya yakni lulus
echo $_POST['lulus'];
?>
```

kemudian kita coba di browser, kita pilih 2 saja misalnya, kemudian submit ... apa yang terjadi ?? Array, ini dikarenakan kita menceklis banyak checkbox, sehingga perlu syntax sendiri untuk dapat menampilkannya, kita akan coba memanggilnya array index 0

```
<?php
// kita ganti dengan [0]
echo $_POST['lulus'][0];
?>
```

kita coba lihat di browser, dan ceklis yang paling pertama itu yang ditampilkan, nah sekarang bagaimana jika kita ingin menampilkan semua yang terpilih atau terceklis ?

```
<?php
// pertama adalah kita menghitung berapa jumlah checkbox yang
// terpilih??
$jumlah_terpilih = count($_POST['lulus']);
// kita bisa tampilkan berapa yang terpilih dengan memanfaatkan echo
echo $jumlah_terpilih ;
?>
```

Silahkan kembali ke browser anda, dan lihat hasilnya ?? oke...

Langkah selanjutnya adalah kita melakukan looping sebanyak ceklis yang terpilih caranya?? buka kembali file tadi

```
<?php
$jumlah_terpilih = count($_POST['lulus']);
// kemudian ganti dengan looping
// lakukan pengulangan mulai dari 0, sampai kurang dari jumlah dari
// ceklis yang terpilih
// kemudian lakukan penambahan satu demi satu sampai pengulangannya
// selesai
for ($x=0;$x<=$jumlah_terpilih;$x++)
{
    // kemudian tampilkan isinya dengan menyebut methode, atribut
    // name, dan arraynya
    echo $_POST['lulus'][$x];
}
?>
```

lalu kita lihat di browser, ceklis yang akan di pilih, kemudian kirim, apa yang terjadi, cukup mudah ya??

5.6 Perbedaan POST dan GET

Apa yang membedakan method post dan get pada suatu form ??? Saya akan coba jelaskan disini, dalam method post seperti praktek-praktek yang sudah kita lakukan sebelumnya, apabila kita lihat url halaman diaddress bar kita lihat. Apabila kita klik kemudian disubmit, kita lihat url address barnya, **process4.php**, url terlihat bersih. dan tidak mengandung parameter-parameter lain selain nama filenya. Apa bedanya dengan GET ? oke langsung saja kita praktikan, kita gunakan file yang sebelumnya **form4_checkbox.html** namun untuk saya sendiri saya melakukan save ulang dengan nama **form5_checkboxget.html**.

```
<html>
<head><title>Form 5 Input Checkbox - Latihan 5</title></head>
<body>

// namun kita ganti methodnya dari POST menjadi GET

<form action="process5.php" method="GET">
<table border='1' width='20%' cellpadding='1' cellspacing='1'
align='center'>
<tr><td>Status Kelulusan : </td><td><input type="checkbox"
name="lulus[]" value="SD"/>SD</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMP"/>SMP
</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMP"/>SMP
</td></tr>
<tr><td></td><td><input type="checkbox" name="lulus[]" value="SMA"/>SMA
</td></tr>
<tr><td></td><td><input type="submit" name="kirim" value="kirim"
/></td></tr>
</table>
</form>
</body>
</html>
```

kemudian di bagian process5.php dengan mengambil isi dari file sebelumnya yakni **process4.php** kita ganti POST menjadi GET juga

```
<?php
$jumlah_terpilih = count($_GET['lulus']);
for ($x=0;$x<$jumlah_terpilih;$x++) {
    echo $_GET['lulus'][$x]."<br /> ";
}
?>
```

Dan kita lihat apa yang terjadi ??? url nya jadi lebih panjang, dan memiliki parameter-paramter yang berisi data yang di inputkan, biasanya get ini digunakan dalam fasilitas search dari suatu website, dan google pun menggunakan method ini dalam pencarian data

...

5.7 Multiple Submit

Multiple submit biasanya digunakan untuk 2 proses yang berbeda, sejauh yang saya temui, biasanya multiple submit digunakan untuk melakukan edit ataupun delete suatu data, bisa juga untuk mempublish ataupun untuk mengedit suatu artikel, dan lain-lain sebagainya ... di sini saya akan coba jelaskan bagaimana menggunakan multiple submit... untuk melakukan dua proses yang berbeda, contoh kasus disini adalah insert dan update,

Ketika input submit insert di klik maka proses insert terjadi, sedangkan jika input submit update di klik maka proses update yang terjadi

Kita akan gunakan file form1.php, dengan sedikit penambahan di sana sini

```
<html>
<head><title>Form 6 - Latihan 5</title></head>
<body>
<form action="process6.php" method="POST">

<table border='1' width='10%' cellpadding='1' cellspacing='1'
align='center'>
<caption><strong>Multiple Submit</strong></caption>
<tr><td>Nama</td><td><input type="text" name="nama" /></td></tr>
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin"
value="laki-laki"/>Laki-laki<br />
<input type="radio" name="jenis_kelamin"
value="perempuan"/>Perempuan</td></tr>
<tr><td>Alamat</td><td><input type="text" name="alamat" /></td></tr>
<tr><td></td><td>

// lalu disini kita akan gunakan dua submit, attribut namanya di isi
// sama, namun valuenya yang berbeda...
<input type="submit" name="kirim" value="update" />
<input type="submit" name="kirim" value="insert" /></td></tr>
</table>
</form>
</body>
</html>
```

kita lihat di browser, terdapat dua submit, kemudian kita akan membuat 2 proses, ketika tombol update di klik apa yang terjadi dan ketika tombol insert di klik apa yang terjadi ...

Oke kita buat satu file dengan nama process6.php, kita akan memanfaatkan

argumen if, else if, dan else, ini kita gunakan untuk kondisional, ketika tombol insert atau tombol update di klik maka ada, sesuatu yang terjadi ...

```
<?php

// jika tombol kirim yang memiliki value insert di klik
if ($_POST['kirim'] == "insert")
{
    echo "Proses Insert...";
}

// namun jika yang di klik itu tombol update
// artinya jika tombol kirim yang memiliki value update di klik
else if ($_POST['kirim'] == "update")
{
    // maka lakukan proses update
    echo "Proses Update...";

    // lalu kita tutup lagi
}

// dan jika tidak ada button yang diklik
else
{
    // maka tidak ada proses apa-apa yang terjadi ...
    echo "Tidak ada proses...";
}

?>
```

kita lihat di browser, ketika button insert di klik, apa yang terjadi, dan ketika tombol update di klik yang terjadi adalah ...

5.8 Dasar Input ke database

Pada bagian ini akan saya jelaskan kepada anda, dasar bagaimana meninsertkan data ke dalam database data yang di input melalui form. ketika admin atau user menginputkan data ke form, ketika tombol submit dikirim data yang dikirim itu akan di proses kemudian di masukkan kedalam database, itu semua akan kita pelajari disini ...

Contoh kasus yang akan saya jelaskan adalah, bagaimana menggunakan form untuk menginsert data. Untuk secara lengkapnya, mengenai bagaimana melakukan insert, delete, update data menggunakan php itu akan saya jelaskan di bab 6. Sementara sub bab ini hanya menerangkan sekilumit saja, mengenai penggunaan form dalam menginsert data.

Oke, disini kita akan membuat form terlebih dahulu, saya sudah menyiapkan filenya di dalam file latihanbab7, dengan nama **form7_insertdatabase.html** :

```
<html>
<head><title>Form 7 - Latihan 5</title></head>
<body>
<form action="process7_database.php" method="POST">
<table border='0' cellpadding='3' cellspacing='3' align='center'>
<caption><strong>Formulir Biodata  
Mahasiswa</strong><br /><br /></caption><br />
<tr><td>Nama</td><td><input type="text" name="nama" /></td></tr>
<tr><td>Jenis Kelamin</td><td><label for="jeniskelamin"><input  
type="radio" name="jeniskelamin" value="pria"/>Pria</label>  
<label for="jeniskelamin"><input type="radio" name="jeniskelamin"  
value="wanita"/>Wanita</label>
</td></tr>
<tr><td>Tanggal lahir</td><td>
<select name="tanggal">
<option value="01">1</option>
<option value="01">1</option>
<option value="02">2</option>
<option value="03">3</option>
<option value="04">4</option>
<option value="05">5</option>
<option value="06">6</option>
<option value="07">7</option>
<option value="08">8</option>
<option value="09">9</option>
<option value="10">10</option>
<option value="11">11</option>
<option value="12">12</option>
<option value="13">13</option>
```

```
<option value="14">14</option>
<option value="15">15</option>
<option value="16">16</option>
<option value="17">17</option>
<option value="18">18</option>
<option value="19">19</option>
<option value="20">20</option>
<option value="21">21</option>
<option value="22">22</option>
<option value="23">23</option>
<option value="24">24</option>
<option value="25">25</option>
<option value="26">26</option>
<option value="27">27</option>
<option value="28">28</option>
<option value="29">29</option>
<option value="30">30</option>
<option value="31">31</option>
</select>

/<select name="bulan">
<option value="01">Januari</option>
<option value="02">Februari</option>
<option value="03">Maret</option>
<option value="04">April</option>
<option value="05">Mei</option>
<option value="06">Juni</option>
<option value="07">Juli</option>
<option value="08">Agustus</option>
<option value="09">September</option>
<option value="10">Oktober</option>
<option value="11">November</option>
<option value="12">Desember</option>
</select>

/<select name="tahun">
<option value="1980">1980</option>
<option value="1981">1981</option>
<option value="1982">1982</option>
<option value="1983">1983</option>
<option value="1984">1984</option>
<option value="1985">1985</option>
<option value="1986">1986</option>
<option value="1987">1987</option>
```

```
<option value="1988">1988</option>
<option value="1989">1989</option>
<option value="1990">1990</option>
<option value="1991">1991</option>
<option value="1992">1992</option>
<option value="1993">1993</option>
<option value="1994">1994</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
</select>
</td></tr>
<tr><td>Alamat</td><td><textarea cols="20" rows="5" name="alamat"></textarea></td></tr>
<tr><td></td><td><input type="submit" name="kirim" value="kirim" /></td></tr>
</table>
</form>
</body>
</html>
```

Kita lihat dulu hasilnya di browser mozilla firefox,

The screenshot shows a web form titled "Formulir Biodata Mahasiswa". The form includes fields for Name (text input), Gender (radio buttons for Pria and Wanita), Date of Birth (dropdown menus for day, month, and year), and Address (text area). A "kirim" button is at the bottom.

Gambar 5.2 (Form Bioadat)

selanjutnya kita membuat file yang dapat memproses inputan tersebut ke dalam database, mula-mula kita buat filenya terlebih dahulu, saya menamakan **process7_database.php**, karena sesuai dengan action pada halaman form yang telah dibuat tadi.

```
<?php
// nama, jeniskelamin, tanggal, bulan, tahun, alamat
// kemudian untuk dapat menginsert databasenya kita kita lakukan
// koneksi dulu
// kita ambil file koneksi yang pada bab3 telah saya jelaskan
// sebelumnya

include "koneksi.php";

// kemudian kita ambil data-data yang dikirim dari formnya
// di masukkan ke dalam variabel
// ini digunakan untuk mempermudah proses insert record ke dalam
// database

$nama = $_POST['nama'];
$jeniskelamin = $_POST['jeniskelamin'];

// untuk dapat menginsertkan tanggal ke dalam database
// terlebih dahulu kita rubah formatnya menjadi tahun-bulan-tanggal
// seperti ini ...
$tanggallahir = $_POST['tahun'].'-'.$_POST['bulan'].'-'
.'$_POST['tanggal'];
$alamat = $_POST['alamat'];

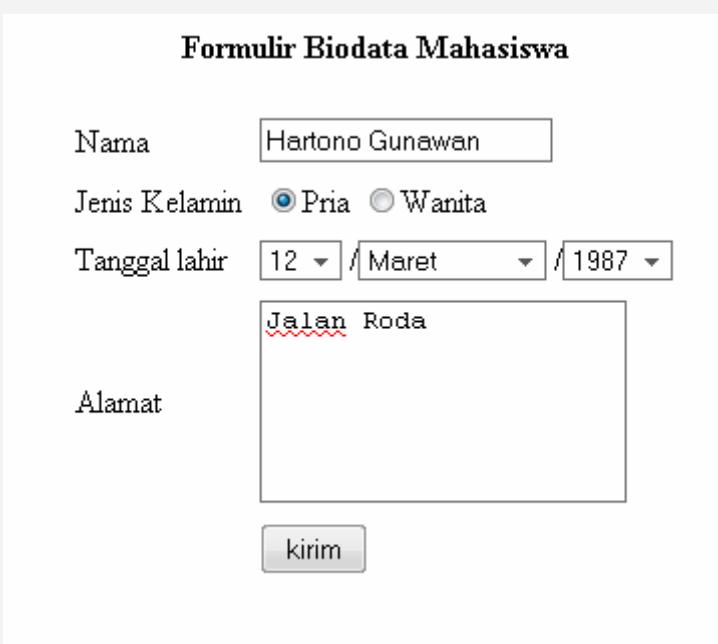
// setelah dari situ kemudian kita buat query mysql untuk insert record
// kedalam tbl_mhsiswa
$insert = "insert into `tbl_mhsiswa` (`nama_mhs`, `jenis_kelamin`,
`tgl_lahir`, `alamat`)
values ('$nama', '$jeniskelamin', '$tanggallahir','$alamat');";

// lalu lakukan querinya
$insert_query = mysql_query($insert);

// untuk memberitahukan jika record yang telah berhasil diinsert ke
// dalam database
if($insert_query) {
echo "Insert Record Berhasil<br />";
echo "Anda Telah berhasil Menginput data:";
echo $nama,"<br />",$jeniskelamin,"<br />",$tanggallahir,"<br
```

```
>",$alamat ; }  
  
// namun jika gagal....  
else  
echo "Gagal Insert Record";  
  
?>
```

oke, beres, kita coba langsung buka formnya di mozilla firefox, kemudian kita inputkan data kedalam form,



Formulir Biodata Mahasiswa

Nama Hartono Gunawan

Jenis Kelamin Pria Wanita

Tanggal lahir 12 / Maret / 1987

Alamat Jalan Roda

Gambar 5.3 (Form Biodata)

kemudian tekan submit, apa yang terjadi,

Insert Record Berhasil
Anda Telah berhasil Menginput data:
Hartono Gunawan
pria
1987-03-12
Jalan Roda

kita lihat di phpmyadmin, data yang telah kita input telah berhasil masuk ke dalam **tbl_mhsiswa** ...

Show: 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None

+ Options

	id_mhs	nama_mhs	jenis_kelamin	tgl_lahir	alamat
<input type="checkbox"/>		X	1 Loka Dwiartara	pria	1987-01-24 Jawa Barat
<input type="checkbox"/>		X	2 Jatrasmara	pria	1985-10-29 Jawa Timur
<input type="checkbox"/>		X	4 Hartono Gunawan	pria	1987-03-12 Jalan Roda

Check All / Uncheck All With selected:

Show: 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

Query results operations

Print view Print view (with full texts) Export CREATE VIEW

Gambar 5.3 (Form Biodata)

5.9 Page Title yang dinamis dan Memanfaatkan URL dalam PHP

Dengan memanfaatkan tag title dalam html dan bantuan php serta mysql, anda dapat membuat halaman html memiliki page title yang dinamis. Di sini pula saya akan jelaskan, bagaimana memanfaat url dalam php dalam memproses inputan dari user. Ada dua langkah yang akan kita kerjakan ...

Pertama kita akan cuma menampilkan nama mahasiswa yang diambil dari database, nama mahasiswa ini akan ditampilkan dalam bentuk url, dan ketika url ini di klik maka akan menampilkan informasi detil mengenai mahasiswa tersebut

Kedua, nama mahasiswa itu akan muncul dalam title dari suatu halaman, yang memberitahukan jika kita sedang mengakses data mengenai mahasiswa bernama si x.

Oke, tahapan pertama adalah kita membuat file untuk menampilkan nama-nama mahasiswa ...

Saya menamakan filenya dengan nama **displaydata.php**

```
<?php
// kita lakukan koneksi terlebih dahulu ...
include "koneksi.php";

// kemudian buat satu query yang akan mengambil id mahasiswa, dan nama
// mahasiswa
// id mahasiswa di ambil sebagai kunci untuk menampilkan detil
// mahasiswa
// jadi detil informasi mahasiswa, berisi nama, alamat, tanggal lahir
// itu akan ditampilkan dengan memanfaatkan idnya nantinya ....
$tampil_nama = "select id_mhs, nama_mhs from tbl_mhsiswa";

// kemudian kita jalankan querynya ...
$tampil_nama_query = mysql_query($tampil_nama);

// lalu kita tampilkan isi databasenya
while ($hasil = mysql_fetch_array($tampil_nama_query))
{
    // lalu kita tampilkan datanya sebagai link url, yang ketika di klik
    // akan menampilkan
    // informasi detil mengenai detil mahasiswa tersebut
    echo "<a href=detaildata.php?id=", $hasil['id_mhs'], ">",
$hasil['nama_mhs'], "</a><br />";
}

?>
```

lalu kita coba tes di browser mozilla firefox, kita lihat urlnya,

```
http://localhost/file.latihan.php/latihanbab5/detaildata.php?id=1
```

Masing-masing memiliki id yang berbeda misalnya id untuk loka dwiartara kita lihat **detaildata.php?id=1**

Dan ketika kita klik nantinya akan menampilkan informasi detil mengenai loka dwiartara ini, tapi untuk sekarang apabila kita klik, ini terjadi karena kita belum memiliki filenya. Kita akan coba buat halaman yang akan dapat menampilkan detil informasi mengenai loka dwiartara sesuai dengan url yang tadi, saya akan membuat file dengan nama **detaildata.php**

```
<?php
// pertama adalah kita lakukan koneksi terlebih dahulu
// antara php dan mysql
include "koneksi.php";

// kemudian kita ambil nilai id yang ada pada url
$id = $_REQUEST['id'];

// lalu kita buat query yang hanya dapat menampilkan informasi
// mahasiswa yang kita klik saja
$detail = "select * from tbl_mhsiswa where id_mhs='$id'";

// kita jalankan querynya
$detail_query = mysql_query($detail);

while ($hasil = mysql_fetch_array($detail_query))
{
    // lalu kita masukkan kedalam variable untuk mempermudah dalam
    // menampilkan data
    $nama = $hasil['nama_mhs'] ;
    $jenis_kelamin = $hasil['jenis_kelamin'];
    $tanggal_lahir = $hasil['tgl_lahir'];
    $alamat = $hasil['alamat'];

    // kemudian kita sisipkan data yang akan ditampilkan dalam satu
    // variabel
    $data_lengkap = "Nama : ".$nama."<br />";
    $data_lengkap .= "Jenis Kelamin : ".$jenis_kelamin."<br />";
    // tanda titik ini berguna untuk menambahkan isi variable yang
```

```
// sebelumnya agar tidak tertimpa
// melainkan hanya menambahkan saja, dan tidak ditampilkan, namun akan
// di tampilkan
// di akhir dari coding halaman ini ...
$data_lengkap .= "Tanggal Lahir : ".$tanggal_lahir."<br />";
$data_lengkap .= "Alamat : ".$alamat."<br />";
}

// kemudian tampilkan semua informasi detil.
echo "Informasi Detil mengenai <strong>$nama</strong> adalah : <br
/>".$data_lengkap;
?>
```

Kita lihat di browser, silahkan di pilih kemudian apa yang terjadi, sudah cukup baik. namun kita lihat page title nya masih statis, dan belum muncul. Bagaimana untuk memunculkan page title sesuai dengan nama mahasiswa yang kita klik ???

Kita cukup menambahkan html tag di bawah **file detailldata.php** tadi ... Isi seperti ini ...

```
<?php
// pertama adalah kita lakukan koneksi terlebih dahulu
// antara php dan mysql
include "koneksi.php";

// kemudian kita ambil nilai id yang ada pada url
$id = $_REQUEST['id'];

// lalu kita buat query yang hanya dapat menampilkan informasi
// mahasiswa yang kita klik saja
$detail = "select * from tbl_mhsiswa where id_mhs='".$id"'";

// kita jalankan querynya
$detail_query = mysql_query($detail);

while ($hasil = mysql_fetch_array($detail_query))
{
// lalu kita masukkan kedalam variable untuk mempermudah dalam
// menampilkan data
$nama = $hasil['nama_mhs'] ;
$jenis_kelamin = $hasil['jenis_kelamin'];
$tanggal_lahir = $hasil['tgl_lahir'];
$alamat = $hasil['alamat'];
```

```
// kemudian kita sisipkan data yang akan ditampilkan dalam satu
// variable
    $data_lengkap = "Nama : ".$nama."<br />";
    $data_lengkap .= "Jenis Kelamin : ".$jenis_kelamin."<br />";

// tanda titik ini berguna untuk menambahkan isi variable yang
// sebelumnya agar tidak tertimpa
// melainkan hanya menambahkan saja, dan tidak ditampilkan, namun akan
// di tampilkan
// di akhir dari coding halaman ini ...
    $data_lengkap .= "Tanggal Lahir : ".$tanggal_lahir."<br />";
    $data_lengkap .= "Alamat : ".$alamat."<br />";
}

// kemudian tampilkan semua informasi detil.
?>

<html>
<head>
<title><?php echo "Informasi $nama"; ?></title>
</head>
<body>
<?php echo "Informasi Detil mengenai <strong>$nama</strong> adalah :
<br />".$data_lengkap; ?>
</body>
</html>
```

coba kita tes di browser, pertama adalah membuka halaman displaydata.php... kita klik, apa yang terjadi, titlenya otomatis berubah ...

Bab 6:

Mengedit Database

dengan Form

6.1 Insert Record

Sebelumnya insert record telah di bahas dalam sub bab di bab sebelumnya, namun di sini saya akan coba jelaskan kembali berguna untuk mengulang materi sebelumnya, tapi saya akan coba bedakan contohnya.

Di sini kita akan coba buat satu buah form yang benar-benar sangat lengkap. Namun sebelumnya kita akan buat database dan tablenya terlebih dahulu. Kita bisa memanfaakan phpmyadmin untuk lebih mudah dalam pembuatan table, tapi bagi anda yang ingin menguasai query sql, saya akan coba jelaskan juga pembuatan tablenya dalam bentuk query sql.

Field yang kita butuhkan adalah sebagai berikut :

field	Type	Size	Attribut
id_daftar	int	10	auto_increment primary key
tanggal_daftar	date		
nama_pendaftar	varchar	75	
jns_kelamin	varchar	15	
lulusan_sekolah	int	5	
tgl_lahir	date		
status	varchar	20	

pekerjaan	varchar	125	
alamat	varchar	150	
kelurahan	varchar	75	
kecamatan	varchar	75	
kota	varchar	75	
provinsi	varchar	75	
telp	varchar	20	

Sehingga apabila kita buat ke dalam bentuk querinya adalah seperti ini, jika anda masih belum mengerti silahkan kembali bab 3, yang membahas mengenai fungsi-fungsi mysql. Apabila anda ingin menggunakan command line (DOS) anda bisa menggunakan query ini :

```
create database mahasiswa_db;
use mahasiswa_db;
create table tbl_daftar_mahasiswa(id_daftar int(10) auto_increment
primary key, tanggal_daftar date, nama_pendaftar varchar(75),
jns_kelamin varchar(15), status varchar(20), lulusan_sekolah
varchar(20), tahun_ajaran int(6), pekerjaan varchar(125), alamat
varchar(150), kelurahan varchar(75), kecamatan varchar(75), kota
varchar(75), provinsi varchar(75), telp varchar(20));
```

Oke kita langsung jalankan saja di cmd, atau anda juga dapat menggunakan phpmyadmin di bagian menu sql (Query Windows)  dalam database mahasiswa_db. Setelah beres, selanjutnya kita akan buat 1 buah form yang dapat menginputkan data-data mahasiswa untuk selanjutnya di masukkan ke dalam database mahasiswa_db.

Pembuatan Form Insert Database ...

Kita akan buat satu buah form yang dapat menginputkan data kemudian di insertkan kedalam database. Terlebih dahulu kita siapkan filenya dengan nama **form.insert.html**

```
<html>
<head><title>Form Insert</title></head>
<body>
<form action="insert.php" method="POST">
<table border='1' width='35%' cellpadding='2' cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2></caption>
```

```
<tr><td>Nama</td><td><input type="text" name="nama" size="30" maxlength="50"/></td></tr>
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin" value="pria"/>
<label for="jenis_kelamin">Pria</label>
<input type="radio" name="jenis_kelamin" value="wanita"/><label for="jenis_kelamin">Wanita</label>
</td></tr>
<tr><td>Status</td><td>
<select name="status">
<option value="Pilih Status Anda">Pilih Status Anda</option>
<option value="menikah">Menikah</option>
<option value="single">Single</option>
</select>
</td></tr>
<tr><td>Lulusan Sekolah</td><td><input type="text" name="sekolah" size="30" maxlength="50"/></td></tr>
<tr><td>Tahun Ajaran</td><td><select name="tahun_ajaran">
<option>Pilih Tahun Ajaran</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
<option value="2005">2005</option>
<option value="2006">2006</option>
<option value="2007">2007</option>
<option value="2008">2008</option>
<option value="2009">2009</option>
<option value="2010">2010</option>
</select></td></tr>
<tr><td>Pekerjaan</td><td><input type="text" name="pekerjaan" size="30" maxlength="50"/></td></tr>
<tr><td>Alamat</td><td><textarea name="alamat" cols="30" rows="4"></textarea></td></tr>
<tr><td>Kelurahan</td><td><input type="text" name="kelurahan" size="30" maxlength="50"/></td></tr>
<tr><td>Kecamatan</td><td><input type="text" name="kecamatan" size="30" maxlength="50"/></td></tr>
```

```
<tr><td>Kota</td><td><input type="text" name="kota" size="30" maxlength="50"/></td></tr>
<tr><td>Provinsi</td><td>
<select name="provinsi">
<option value="Pilih Provinsi Anda">Pilih Provinsi Anda</option>
<option value="Bali">Bali</option>
<option value="Bangka Belitung">Bangka Belitung</option>
<option value="Banten">Banten</option>
<option value="Bengkulu">Bengkulu</option>

<option value="Gorontalo">Gorontalo</option>
<option value="Irian Jaya Barat">Irian Jaya Barat</option>
<option value="Jabotabekdecipsawcib">Jabotabekdecipsawcib</option>
<option value="Jambi ">Jambi</option>
<option value="Jawa Barat">Jawa Barat</option>
<option value="Jawa Tengah">Jawa Tengah</option>
<option value="Jawa Timur">Jawa Timur</option>
<option value="Kalimantan Barat">Kalimantan Barat</option>
<option value="Kalimantan Selatan">Kalimantan Selatan</option>

<option value="Kalimantan Tengah">Kalimantan Tengah</option>
<option value="Kalimantan Timur">Kalimantan Timur</option>
<option value="Kepulauan Riau">Kepulauan Riau</option>
<option value="Lampung">Lampung</option>
<option value="Lokasi Lain-lain">Lokasi Lain-lain</option>
<option value="Maluku">Maluku</option>
<option value="Maluku Utara">Maluku Utara</option>
<option value="Nangroe Aceh Darussalam">Nangroe Aceh Darussalam</option>
<option value="Nusa Tenggara Barat">Nusa Tenggara Barat</option>

<option value="Nusa Tenggara Timur">Nusa Tenggara Timur</option>
<option value="Papua">Papua</option>
<option value="Riau">Riau</option>
<option value="Sulawesi Barat">Sulawesi Barat</option>
<option value="Sulawesi Selatan">Sulawesi Selatan</option>
<option value="Sulawesi Tengah">Sulawesi Tengah</option>
<option value="Sulawesi Tenggara">Sulawesi Tenggara</option>
<option value="Sulawesi Utara">Sulawesi Utara</option>
<option value="Sumatera Barat">Sumatera Barat</option>

<option value="Sumatera Selatan">Sumatera Selatan</option>
<option value="Sumatera Utara">Sumatera Utara</option>
<option value="Yogyakarta">Yogyakarta</option>
```

```

</select></td></tr>
<tr><td>No Telepon</td><td><input type="text" name="telepon" size="30" maxlength="50"/></td></tr>
<tr><td></td><td><input type="submit" name="kirim" value="Daftar! "/></td></tr>
</table>
</form>
</body>
</html>

```

Setelah itu kita coba tes di browser, isi formulirnya kemudian tekan submit, dan ups... karena file **insert.php** belum di buat maka tampilannya pun seperti itu, oleh karena itu terlebih dahulu kita balik ke editor notepad++, kemudian disini kita ketikkan variable-variable yang telah dikirim melalui **form.insert.html**.

Oke, kita buat file **insert.php**, kemudian untuk mengetahui apakah form yang telah di buat tadi sudah benar-benar mengirim semua variablenya kedalam **insert.php** caranya adalah dengan menampilkan semua variable yang sudah di pos, seperti ini :

```

<?php
echo $_POST['nama']."<br/>" ;
echo $_POST['jenis_kelamin']."<br/>" ;
echo $_POST['status']."<br/>" ;
echo $_POST['sekolah']."<br/>" ;
echo $_POST['tahun_ajaran']."<br/>" ;
echo $_POST['pekerjaan']."<br/>" ;
echo $_POST['alamat']."<br/>" ;
echo $_POST['kelurahan']."<br/>" ;
echo $_POST['kecamatan']."<br/>" ;
echo $_POST['kota']."<br/>" ;
echo $_POST['provinsi']."<br/>" ;
echo $_POST['telepon']."<br/>" ;
?>

```

Kemudian kita cek dengan mengisi kembali form inputan yang ada ... ya lakukan pengisian... kemudian tekan submit, dan hasilnya. Oke setelah semuanya berhasil diinputkan dan juga ditampilkan dalam **insert.php**, langkah selanjutnya yang harus kita lakukan adalah, menginsertkan ke database semua yang di inputkan tersebut, kita kembali kedalam file **insert.php**, langkah selanjutnya adalah kita coba insertkan ke database dengan catatan sebelum dapat menginsertkan ke dalam database kita terlebih dahulu koneksi antara si php dengan mysqlnya ...

Pertama kita buat terlebih dahulu file konesinya, dengan nama **koneksi.php**

```
<?php
$host = "localhost";
$username = "root";
$password = "";
$database = "mahasiswa_db";

$koneksi = mysql_connect($host, $username, $password);
$pilihdatabase = mysql_select_db($database, $koneksi);
?>
```

Lalu selanjutnya kita langsung buat file **insert.php**, diletakkan satu direktori dengan file **koneksi.php**

```
<?php
// kita konesikan dulu dengan file koneksi.php
include "koneksi.php";

// kemudian kita buat variable-variable yang akan kita masukkan ke
// dalam database nanti
// diambil dari form inputan ...
$tanggal_daftar = date("Y-n-d");
$nama = $_POST['nama'];
$jenis_kelamin = $_POST['jenis_kelamin'];
$status = $_POST['status'];
$sekolah = $_POST['sekolah'];
$tahun_ajaran = $_POST['tahun_ajaran'];
$pekerjaan = $_POST['pekerjaan'];
$alamat = $_POST['alamat'];
$kelurahan = $_POST['kelurahan'];
$kecamatan = $_POST['kecamatan'];
$kota = $_POST['kota'];
$provinsi = $_POST['provinsi'];
$telepon = $_POST['telepon'];

// buat query untuk memasukkan hasil inputan tadi ke dalam database
$query_insert = "insert into tbl_daftar_mahasiswa(tanggal_daftar,
nama_pendaftar, jns_kelamin, status, lulusan_sekolah, tahun_ajaran,
pekerjaan, alamat, kelurahan, kecamatan, kota, provinsi, telp)
values('$tanggal_daftar', '$nama', '$jenis_kelamin', '$status',
'$sekolah', '$tahun_ajaran', '$pekerjaan', '$alamat',
'$kelurahan', '$kecamatan', '$kota', '$provinsi','$telepon');";

// lalu jalankan querynya untuk langsung dapat memasukkannya kedalam
```

```
// database.

$insert = mysql_query($query_insert);

// apabila query insert berjalan dengan baik,
if($insert)
{
// maka
    echo "Data anda telah berhasil di input ke dalam database ...";
}

// jika querynya gagal, atau ada yang tidak beres dengan koneksi antara
// php dan mysql

else
{
// maka
    echo "Gagal insert database ... ";
}

?>
```

Oke setelah semua beres, yang perlu kita lakukan adalah mencobanya kembali untuk mengisi form inputan yang tadi. Masuk ke dalam browser ... dan kita isi ... lalu kita kirim ... dan hasilnya... untuk membuktikan record dalam tbl_daftar_mahasiswa telah terisi kita akan coba buka dalam phpmyadmin. Record telah berhasil di inputkan ke dalam database...

Saya mengimbau kepada anda untuk memperhatikan secara teliti benar salahnya query yang akan digunakan dalam php untuk memanipulasi mysql database, karena jika terjadi kesalahan semua yang diinputkan melalui form itu tidak akan bisa di insertkan kedalam database.

Oke setelah perintah insert telah berhasil, kita coba untuk menampilkan yang ada dalam database dengan menggunakan html table. akan saya jelaskan di subbab berikutnya.

6.2 Select Record

select record, atau menampilkan isi dari suatu table, sama halnya dengan insert query, untuk dapat menampilkan isi table menggunakan php, terlebih dahulu anda harus mengkoneksikan antara php dengan mysql terlebih dahulu. select record benar-benar sangat bermanfaat kegunaannya, ketika anda menginputkan suatu artikel diblog dengan si programmer menggunakan perintah insert untuk dapat menginputkan artikel yang anda posting di blog, kemudian dengan select lah artikelnya dapat ditampilkan di halaman index blog anda, atau di dalam kategori, kategori, sehingga jelasnya insert dan select merupakan perintah dasar yang benar-benar sangat bermanfaat dalam membuat aplikasi berbasis website menggunakan php dan mysql.

oke, sebelumnya kita telah menginputkan data menggunakan form inputan ke dalam database, selanjutnya kita akan coba menampilkan isi dari database tersebut. caranya adalah menggunakan select, kita buat filenya terlebih dahulu ...

dengan nama **select.php**, kita buat filenya 1 direktori dengan **koneksi.php** yang telah dibuat sebelumnya...

```
<?php
// kita lakukan koneksi dulu seperti biasanya ...
include "koneksi.php";

// buat query yang dapat menampilkan semua isi tbl_daftar_mahasiswa
$select = "select * from tbl_daftar_mahasiswa order by id_daftar desc";

// jadikan variable $select itu menjadi query yang siap di eksekusi
// oleh mysql
$select_query = mysql_query($select);
?>

<html>
<head>
<title>Latihan BAB 6 - Select Record</title>
</head>
<body>

// pembuatan table sebelumnya telah saya jelaskan...
<table style="font-size:11px;font-family:'arial';text-align:center;" border='1' width="80%" align='center' >
<caption><h1>Daftar Calon Mahasiswa Baru</h1></caption>
```

```

<tr><th>No</th><th>Tanggal</th><th>Nama</th><th>Jenis
Kelamin</th><th>Status</th><th>Lulusan</th><th>Tahun</th>
<th>Pekerjaan</th><th>Alamat</th><th>Kelurahan</th><th>Kecamatan</th><t
h>Kota</th><th>Provinsi</th><th>Telepon</th>
<th>Action</th></tr>

<?php

// kemudian kita tampilkan isi yang berada dalam database tadi ...
while($select_result = mysql_fetch_array($select_query))
{
    // kita ambil saatu persatu field yang berada dalam
tbl_daftar_mahasiswa tadi ...
$id = $select_result['id_daftar'] ;
$tanggaldafatar = $select_result['tanggal_daftar'] ;
$nama = $select_result['nama_pendaftar'] ;
$jeniskelamin = $select_result['jns_kelamin'] ;
$status = $select_result['status'] ;
$lulusan = $select_result['lulusan_sekolah'] ;
$tahunajaran = $select_result['tahun_ajaran'] ;
$pekerjaan = $select_result['pekerjaan'] ;
$alamat = $select_result['alamat'] ;
$kelurahan = $select_result['kelurahan'] ;
$kecamatan = $select_result['kecamatan'] ;
$kota = $select_result['kota'] ;
$provinsi = $select_result['provinsi'] ;
$telepon = $select_result['telp'] ;

// lalu kita tampilkan dalam bentuk table yang dinamis ke bawah
// caranya adalah dengan menampilkan isi variable didalam tag-tag td

echo "<tr><td>$id</td><td>$tanggaldafatar</td><td>$nama
</td><td>$jeniskelamin</td><td>$status</td><td>$lulusan</td>
<td>$tahunajaran</td><td>$pekerjaan</td><td>$alamat</td><td>$kelurahan<
/td><td>$kecamatan</td><td>$kota</td>
<td>$provinsi</td><td>$telepon</td>

// lalu tutup dengan tag tr ...
</tr>";
}

?>
```

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

Oke sudah semua, kemudian kita jalankan file tersebut menggunakan browser ...

6.3 Delete Record

Oke, aksi selanjutnya yang akan kita buat adalah aksi delete. delete disini nantinya akan digunakan untuk menghapus record yang dipilih... sebelumnya di bab 3 saya telah menjelaskan syntax dasar dalam mendelete record, namun di sini kita akan coba memperdalam delete record ... kita masih menggunakan file yang sama yakni select.php, namun kita akan coba menambahkan form di setiap barisnya untuk dapat menjalankan aksi delete dan update record.

```
<?php
include "koneksi.php";
$select = "select * from tbl_daftar_mahasiswa order by id_daftar desc";
$select_query = mysql_query($select);
?>

<html>
<head>
<title>Latihan BAB 6 - Select Record</title>
</head>
<body>

<table style="font-size:11px;font-family:'arial';text-align:center;" border='1' width="80%" align='center' >
<caption><h1>Daftar Calon Mahasiswa Baru</h1></caption>
<tr><th>No</th><th>Tanggal</th><th>Nama</th><th>Jenis Kelamin</th><th>Status</th><th>Lulusan</th><th>Tahun</th><th>Pekerjaan</th><th>Alamat</th><th>Kelurahan</th><th>Kecamatan</th><th>Kota</th><th>Provinsi</th><th>Telepon</th><th>Action</th></tr>

<?php
while($select_result = mysql_fetch_array($select_query))
{
$id = $select_result['id_daftar'] ;
$stanggaldaftar = $select_result['tanggal_daftar'] ;
$nama = $select_result['nama_pendaftar'] ;
$jeniskelamin = $select_result['jns_kelamin'] ;
$status = $select_result['status'] ;
$lulusan = $select_result['lulusan_sekolah'] ;
$tahunajaran = $select_result['tahun_ajaran'] ;
$pekerjaan = $select_result['pekerjaan'] ;
$alamat = $select_result['alamat'] ;
```

```

$kelurahan = $select_result['kelurahan'] ;
$kecamatan = $select_result['kecamatan'] ;
$kota = $select_result['kota'] ;
$provinsi = $select_result['provinsi'] ;
$telepon = $select_result['telp'] ;

echo "<tr><td>$id</td><td>$tanggaldafatar</td><td>$nama</td>
<td>$jeniskelamin</td><td>$status</td><td>$lulusan</td>
<td>$tahunajaran</td><td>$pekerjaan</td>
<td>$alamat</td><td>$kelurahan</td><td>$kecamatan</td><td>$kota</td>
<td>$provinsi</td><td>$telepon</td>
<td>

// kemudian kita siapkan sebuah form yang nantinya digunakan untuk
// menjalankan aksi update dan delete
// kita buat actionnya itu menuju url yang nanti url tersebut digunakan
// untuk variable untuk
// membantu aksi selanjutnya ...
// lihat di sebelah sini ... hasilnya nanti adalah action.php?id=xx ...
// nah id xx inilah yang nantinya
// akan kita gunakan untuk membantu aksi selanjutnya seperti proses
// delete, atau pun edit ...

<form method=\"POST\" action=\"action.php?id=$id\"><input
type=\"hidden\" value=\"$id\" name=\"id\">

// kita buat button untuk update
<input type=\"submit\" name=\"action\" value=\"Update!\"><br />

// dan kita buat button untuk delete recordnya
<input type=\"submit\" name=\"action\" value=\"Delete!\">
</form>
</td>
// lalu tutup dengan tag tr ...
</tr>";
}

?>

</table>
</body>
</html>

```

setelah beres semua formnya, langkah selanjutnya adalah membuat halaman action.php,

Bab 6: Mengedit Database dengan Form

Oleh : Loka Dwiartara | <http://www.ilmuwebsite.com>

halaman **action.php** digunakan untuk mendelete ataupun mengupdate record, dalam sub bab ini saya hanya akan menjelaskan bagaimana melakukan delete record yang dipilih / disubmit kan tersebut, di bagian selanjutnya baru akan saya jelaskan bagaimana cara mengupdate recordnya ...

Oke, kita akan buat **file action.php** ...

Di sini kita akan menggunakan url hasil dari table sebelumnya yakni :

```
http://localhost/latihanbab6/action.php?id=x
```

Si x ini lah yang akan kita manfaatkan, mengaksesnya dengan `$_REQUEST[]` langsung saja ...

```
<?php
// pertama kita ambil aksinya, hal ini diambil berdasarkan dari action
// form sebelumnya
// kemudian kita jadikan huruf kecil semua dengan strtolower()
$action = strtolower($_POST['action']);

// kita ambil url id nya dengan
$id = $_REQUEST['id'];

// lalu kita cocokkan ketika tombol delete yang ditekan
if ($action == "delete")
{
    // maka
    echo "Anda memilih aksi delete pada record $id...";
}

// jika bukan tombol delete yang ditekan
else
{
    // maka
    echo "Anda memilih aksi update...";
}
?>
```

Kita lihat di browser. Kemudian kita klik tombol delete pada salah satu baris, lalu apa yang terjadi? Kita kembali ke notepad lalu kita selesaikan querynya ...

```
<?php
// lakukan koneksi terlebih dahulu
include "koneksi.php";
// kemudian ambil aksi dari form
$action = strtolower($_POST['action']);
// lalu ambil id dari url
$id = $_REQUEST['id'];

if ($action == "delete")
{
    echo "Anda memilih aksi delete pada record $id... <br/>";
    // buat querynya untuk mendelete record mahasiswa yang memiliki
    // id yang akan di delete
    $delete = "delete from tbl_daftar_mahasiswa where id_daftar =
        '$id'";
    // jalankan querynya
    $delete_query = mysql_query($delete);

    // jika querynya berhasil maka record berhasil di hapus
    if ($delete_query) {echo "Record $id berhasil dihapus ...";}
    else {echo "Gagal menghapus record";}

}
else
{
    echo "Anda memilih aksi update...";
}
?>
```

Kita jalankan di browser dan ... ???

6.4 Update Record

Di subbab ini saya menjelaskan bagaimana melakukan update database, masih menggunakan contoh yang sama yaitu menggunakan database **mahasiswa_db** dan **tbl_daftar_mahasiswa**. di dalam file **action.php** kita akan memberikan sedikit sentuhan untuk dapat melakukan update record yang dipilih. Jika record update di pilih maka tampilkan form, oleh karena itu di bagian baris yang berisi kalimat :

```
echo "Anda memilih aksi update...";
```

Di tambahkan menjadi ...

```
<?php
// kita includekana koneksi.php untuk menampilkan record yang akan
// diupdate

include "koneksi.php";
$action = strtolower($_POST['action']);
$id = $_REQUEST['id'];

if ($action == "delete")
{
echo "Anda memilih aksi delete pada record $id... <br/>";
$delete = "delete from tbl_daftar_mahasiswa where id_daftar = '$id'";
$delete_query = mysql_query($delete);
if ($delete_query) echo "Record $id berhasil dihapus ...";
else echo "Gagal menghapus record";
}

else
{

// buat query untuk menampilkan record yang dipilih ...

$select = "select * from tbl_daftar_mahasiswa where id_daftar = '$id'";
$select_query = mysql_query($select);

// lalu sisipkan field yang akan ditampilkan ke dalam variable...
while($select_result = mysql_fetch_array($select_query))
{
    $id = $select_result['id_daftar'] ;
    $nama = $select_result['nama_pendaftar'] ;
```

```

$jeniskelamin = $select_result['jns_kelamin'] ;
$status = $select_result['status'] ;
$lulusan = $select_result['lulusan_sekolah'] ;
$tahunajaran = $select_result['tahun_ajaran'] ;
$pekerjaan = $select_result['pekerjaan'] ;
$alamat = $select_result['alamat'] ;
$kelurahan = $select_result['kelurahan'] ;
$kecamatan = $select_result['kecamatan'] ;
$kota = $select_result['kota'] ;
$provinsi = $select_result['provinsi'] ;
$telepon = $select_result['telp'] ;

// kita tutup terlebih dahulu tagnya, karena kita akan membuat form
// html
?>

// kita buat formnya disini ...
// form update dan insert tidak terlalu berbeda,
// yang jadi perbedaan adalah form update itu memiliki value yang
// isinya itu
// di ambil dari database ...

<form action="update.php" method="POST">
<table border='0' width='35%' cellpadding='2' cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2></caption>

// sengaja saya buat input dengan type hidden
// hidden tidak akan tampil, namun nanti akan kita gunakan
// dalam mengupdate record yang dipilih ...
// dan valuenya kita isi dengan mengambil dari database yakni id_daftar
// yang disisipkan kedalam
// variable id

<input type="hidden" name="id" size="30" value=<?php echo $id ; ?>
maxlength="50"/>

// begitu juga dengan nama
// kita tambahkan valuenya dengan mengambil isinya dari database

<tr><td>Nama</td><td><input type="text" name="nama" value=<?php echo
$nama ; ?>" size="30" maxlength="50"/></td></tr>
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin"
value="pria"/><label for="jenis_kelamin">Pria</label>

```

```

<input type="radio" name="jenis_kelamin" value="wanita"/><label
for="jenis_kelamin">Wanita</label>
</td></tr>
<tr><td>Status</td><td>
<select name="status">
<option value="Pilih Status Anda">Pilih Status Anda</option>
<option value="menikah">Menikah</option>
<option value="single">Single</option>
</select>
</td></tr>
<tr><td>Lulusan Sekolah</td><td><input type="text" name="sekolah"
value=<?php echo $lulusan ; ?>" size="30" maxlength="50"/></td></tr>
<tr><td>Tahun Ajaran</td><td><select name="tahun_ajaran">
<option>Pilih Tahun Ajaran</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
<option value="2005">2005</option>
<option value="2006">2006</option>
<option value="2007">2007</option>
<option value="2008">2008</option>
<option value="2009">2009</option>
<option value="2010">2010</option>
</select></td></tr>
<tr><td>Pekerjaan</td><td><input type="text" name="pekerjaan"
value=<?php echo $pekerjaan ; ?>" size="30" maxlength="50"/></td></tr>
<tr><td>Alamat</td><td><textarea name="alamat" cols="30"
rows="4"><?php echo $alamat ; ?></textarea></td></tr>
<tr><td>Kelurahan</td><td><input type="text" name="kelurahan"
value=<?php echo $kelurahan ; ?>" size="30" maxlength="50"/></td></tr>
<tr><td>Kecamatan</td><td><input type="text" name="kecamatan"
value=<?php echo $kecamatan; ?>" size="30" maxlength="50"/></td></tr>
<tr><td>Kota</td><td><input type="text" name="kota" size="30"
value=<?php echo $kota ; ?>" maxlength="50"/></td></tr>
<tr><td>Provinsi</td><td>
<select name="provinsi">
<option value="Pilih Provinsi Anda">Pilih Provinsi Anda</option>

```

```
<option value="Bali">Bali</option>
<option value="Bangka Belitung">Bangka Belitung</option>
<option value="Banten">Banten</option>
<option value="Bengkulu">Bengkulu</option>

<option value="Gorontalo">Gorontalo</option>
<option value="Irian Jaya Barat">Irian Jaya Barat</option>
<option value="Jabotabekdecipsawcib">Jabotabekdecipsawcib</option>
<option value="Jambi">Jambi</option>
<option value="Jawa Barat">Jawa Barat</option>
<option value="Jawa Tengah">Jawa Tengah</option>
<option value="Jawa Timur">Jawa Timur</option>
<option value="Kalimantan Barat">Kalimantan Barat</option>
<option value="Kalimantan Selatan">Kalimantan Selatan</option>

<option value="Kalimantan Tengah">Kalimantan Tengah</option>
<option value="Kalimantan Timur">Kalimantan Timur</option>
<option value="Kepulauan Riau">Kepulauan Riau</option>
<option value="Lampung">Lampung</option>
<option value="Lokasi Lain-lain">Lokasi Lain-lain</option>
<option value="Maluku">Maluku</option>
<option value="Maluku Utara">Maluku Utara</option>
<option value="Nangroe Aceh Darussalam">Nangroe Aceh
Darussalam</option>
<option value="Nusa Tenggara Barat">Nusa Tenggara Barat</option>

<option value="Nusa Tenggara Timur">Nusa Tenggara Timur</option>
<option value="Papua">Papua</option>
<option value="Riau">Riau</option>
<option value="Sulawesi Barat">Sulawesi Barat</option>
<option value="Sulawesi Selatan">Sulawesi Selatan</option>
<option value="Sulawesi Tengah">Sulawesi Tengah</option>
<option value="Sulawesi Tenggara">Sulawesi Tenggara</option>
<option value="Sulawesi Utara">Sulawesi Utara</option>
<option value="Sumatera Barat">Sumatera Barat</option>

<option value="Sumatera Selatan">Sumatera Selatan</option>
<option value="Sumatera Utara">Sumatera Utara</option>
<option value="Yogyakarta">Yogyakarta</option>
</select></td></tr>
<tr><td>No Telepon</td><td><input type="text" name="telepon" size="30"
value="<?php echo $telepon ; ?>" maxlength="50"/></td></tr>
<tr><td></td><td><input type="submit" name="kirim"
value="Update!" /></td></tr>
```

```
</table>
</form>

<?php
    }
}

?>
```

Setelah semua beres, selanjutnya kita tinggal eksekusi dengan memilih tombol update lalu kita coba lakukan update dan ?? Mantap!

Bab 7:

Lebih Jauh dengan Form

7.1 Validasi Hasil Input

Di bagian ini saya akan jelaskan bagaimana cara seorang web developer itu mengecek hasil inputan yang di lakukan oleh user, sebagai contohnya misalkan user akan melakukan pendaftaran sebagai mahasiswa baru, namun form di bagian nomor teleponnya itu di kosongkan, sedangkan syarat untuk mendaftar adalah, bagian nomor telepon itu harus di isi, nah bagaimana menangani ini sehingga nanti jika ada user yang mengosongkan nomor telepon nya si user tersebut diperintahkan kembali untuk mengisi form bagian nomor telepon.

Atau contoh lainnya misalkan ketika user menginputkan namanya pada nomor telepon, hal ini tentu saja salah, namun bagaimana cara untuk menangani ini, jadi ketika user memasukkan nama pada inputan nomor telepon akan ditolak dan diberikan warning, jika yang diinputkan itu salah. Saya akan coba jelaskan contoh kasus yang pertama, ketika kondisi si user itu mengosongkan salah satu bagian form. untuk mengetahui kosong atau tidaknya suatu variable itu diwakilkan dengan fungsi !empty(). Penulis akan coba jelaskan contoh kasusnya, pertama kita buat form nya terlebih dahulu :

```
<html>
<head><title>Form Insert</title></head>
<body>

<form action="insert.php" method="POST">
<table border='0' width='35%' cellpadding='2' cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2><p><?php echo
$warning ; ?></p>
```

```

</caption>

<tr><td>Nama</td><td><input type="text" name="nama" size="30"
maxlength="50"/></td></tr>
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin"
value="pria"/><label for="jenis_kelamin">
Pria</label>
<input type="radio" name="jenis_kelamin" value="wanita"/><label
for="jenis_kelamin">Wanita</label>
</td></tr>
<tr><td>Alamat</td><td><textarea name="alamat" cols="30"
rows="4"></textarea></td></tr>
<tr><td>No Telepon</td><td><input type="text" name="telepon" size="30"
maxlength="50"/></td></tr>
<tr><td></td><td><input type="submit" name="kirim"
value="Daftar!" /></td></tr>
</table>
</form>
</body>
</html>

```

Ini form yang terdiri dari nama, jenis kelamin, alamat, dan nomor telepon. Kita lihat di browser selanjutnya kita akan buat untuk bagian validasinya, yakni file **insert.php**.

```

<?php
// jika user mengosongkan nama
// ini juga berlaku jika anda mengosongkan semuanya ...
// yang pertama kali di cek adalah bagian ini ...

if (empty($_POST['nama']))
{
    echo "Anda mengosongkan Nama...";
}

// jika user mengosongkan jenis kelamin
else if (empty($_POST['jenis_kelamin']))
{
    echo "Anda mengosongkan jenis kelamin...";
}

// jika user mengosongkan alamat
else if (empty($_POST['alamat']))
{
    echo "Anda mengosongkan alamat";
}

```

```
}

// ini berarti mengosongkan nomor telepon
else if (empty($_POST['telepon']))
{
    echo "Anda mengosongkan no telepon ... ";
}

else
{
    echo "Semua terisi dan proses input ke database bisa di lakukan
disini ... ";
}
?>
```

lalu kita cek di browser dengan mengisi formnya, dan kita klik submit, dan apa yang terjadi ...

7.2 Memfilter Format yang salah

Contoh dalam memfilter format yang salah bisa kita temui ketika kita mengisi form registrasi di bagian inputan untuk email, ketika kita salah mengetik format email dengan salah, misalnya namaemail@yahooocom tanpa tanda titik setelah yahoo, maka akan ada warning jika format email yang kita masukkan itu salah, ada juga contoh kasus lain misalnya ketika kita salah mengisi tanggal lahirnya, ada situs yang memberikan format seperti ini 24-01-1987, namun kemudian kita mengisinya dengan 24-Januari-1987, maka akan ada warning jika format yang diisi itu salah. Solusinya adalah kita menggunakan regular expression, saya hanya menjelaskan sedikit saja regular expression sesuai dengan contoh kasus, karena untuk regular expression itu sendiri butuh satu bahasan terperinci saking banyaknya sub sub dalam regular expression.

Regular expression merupakan suatu pola yang dapat mencocokan kalimat sesuai dengan format yang kita inginkan. Sebelum beranjak ke regular expression kita akan coba terlebih dahulu membuat form dan mengecek format data yang diinputkan, apakah data yang diinput itu berupa angka atau numeric, huruf atau string, apakah data yang diinput itu sebuah array?? Kita buat form terlebih dahulu :

```
<html>
<head><title>Form Check Input</title></head>
<body>

<form action="check_input.php" method="POST">
<table border='0' width='35%' cellpadding='2' cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2><p><?php echo
$warning ; ?></p>
</caption>

<!-- di sini saya coba membuat 6 inputan, yang isinya ada yang terdiri
dari huruf saja dan ada yang terdiri angka saja -->

<tr><td>Nama</td><td><input type="text" name="nama" size="30"
maxlength="50" /></td></tr>
<tr><td>Jenis Kelamin</td><td><input type="radio" name="jenis_kelamin"
value="pria"/><label for="jenis_kelamin">Pria</label>
<input type="radio" name="jenis_kelamin" value="wanita"/><label
for="jenis_kelamin">Wanita</label>
</td></tr>
<tr><td>Tanggal Lahir</td><td><input type="text" name="tanggal_lahir"
size="30" maxlength="50" /></td></tr>
```

```

<tr><td>Alamat</td><td><textarea name="alamat" cols="30" rows="4"></textarea></td></tr>
<tr><td>No Telepon</td><td><input type="text" name="telepon" size="30" maxlength="50" /></td></tr>
<tr><td></td><td><input type="submit" name="kirim" value="Daftar!" /></td></tr>
</table>
</form>
</body>
</html>

```

Kemudian selanjutnya kita akan membuat satu buah file yang dapat mengecek, format data yang diinputkan tersebut apakah angka saja atau huruf saja atau terdapat kedua-duanya. Hal ini menentukan untuk proses selanjutnya. Kita buat satu halaman dengan nama **check_input.php**

```

<?php
// kita cuma akan mengambil sampel untuk nama dan telepon saja...
// selebihnya bisa anda praktikan nanti

$nama = $_POST['nama'];
$telepon = $_POST['telepon'];

// jika nama yang di inputkan itu terdiri dari angka saja
if (is_numeric($nama))
{
    // maka berikan warning jika yang di masukkan itu bertipe numeric...
    echo "<strong>Nama yang anda masukkan bertipe numeric</strong>,
silahkan isi kembali dengan string...<br />";
}

// dan apabila nama yang diinputkan itu bertipe string
else if (is_string($nama))
{
    echo "<strong>Nama yang anda masukkan bertipe string </strong>
<br />";
}

else
{
    echo "Anda harus memasukkan nama dalam format string...<br />";
}

```

```

// begitu juga dengan nomor telepon, apabila yang diinputkan itu
// bertipe numeric
if (is_numeric($telepon))
{
// maka ...
echo "<strong>Nomor Telepon</strong> yang anda masukkan <strong>
bertipe numeric</strong><br />";
}

else
{
// maka
echo "Anda harus memasukkan nama dalam format numeric...<br />";
}
?>

```

Regular expression

Penggunaan regular expresion sebagai contohnya di temui dalam pengisian tanggal lahir, mencocokan format tanggal lahir, sehingga inputan yang di masukkan oleh user nanti bisa seragam mengikuti format yang benar. seperti yang saya katakan sebelumnya, misalkan format yang kita inginkan ketika user menginput tanggal adalah 24-01-1987, ketika ada user yang menginput 24-Januari-1987 maka akan terdapat peringatan jika format yang di masukkan itu adalah salah. Solusinya adalah kita menggunakan regular expression, pertama kita akan buat formnya terlebih dahulu ...

```

<html>
<head><title>Form Check Input</title></head>
<body>

<form action="regex1.php" method="POST">
<table border='0' width='35%' cellpadding='2'  cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2><p><?php echo
$warning ; ?></p>
</caption>

<tr><td>Tanggal Lahir</td><td><input type="text" name="tanggal_lahir"
size="30" maxlength="50" /></td></tr>

```

```

<tr><td></td><td><td><input type="submit" name="kirim"
value="Daftar!" /></td></tr>
</table>
</form>
</body>
</html>

```

kemudian file **regex1.php**

```

<?php
// kita ambil variable yang di ambil dari form inputan, yakni
// tanggallahir
$tanggallahir = $_POST['tanggal_lahir'];

// kemudian kita buat pola dari tanggal lahir
// apabila kita lihat di sini
// ([0-9]{2})-([0-9]{2})-([0-9]{4})

// ([0-9]{2})
// kita tahu bahwa tanggal itu memiliki 2 digit yang isinya terdiri
// dari angka 0 - 9

// ([0-9]{2})
// kita tahu bahwa bulan itu memiliki 2 digit yang isinya terdiri dari
// angka 0 - 9

// ([0-9]{4})
// dan untuk tahun itu memiliki 4 digit angka yang terdiri dari angka 0
// - 9

// dan tanda -
// itu membuat tanggalnya nanti memiliki bentuk seperti ini ... xx-xx-
// xxxx

if (!ereg ("([0-9]{2})-([0-9]{2})-([0-9]{4})", $tanggallahir,
$bagiantanggal))
{
    // perintah !ereg digunakan untuk mencocokan tanggallahir
    // apakah sudah sesuai dengan formatnya
    // yang ditentukan
    // dan variable $bagiantanggal ini digunakan untuk
    // memetakan tanggal lahir yang di masukkan ...
}

```

```

        echo "Format yang anda masukkan salah ...";
    }

else
{
    // sehingga hasilnya nanti seperti ini ...
    echo "Tanggal : $bagiantanggal[1]<br />";
    echo "Bulan : $bagiantanggal[2]<br />";
    echo "Tahun : $bagiantanggal[3]<br />";
}

?>

```

Kita coba lakukan testing di browser ... dan hasilnya ... begitu juga dengan format email kita buat terlebih dahulu formnya

```

<html>
<head><title>Form Check Input</title></head>
<body>

<form action="regex2.php" method="POST">
<table border='0' width='35%' cellpadding='2' cellspacing='2'
align='center'>
<caption><h2>Formulir Pendaftaran Mahasiswa Baru</h2><p><?php echo
$warning ; ?></p>
</caption>
<tr><td>Email</td><td><input type="text" name="email" size="30"
maxlength="50" /></td></tr>
<tr><td></td><td><input type="submit" name="kirim"
value="Daftar!" /></td></tr>
</table>
</form>
</body>
</html>

```

Kemudian kita proses email yang di masukkan dalam file **regex2.php** ...

```

<?php
$email = $_POST['email'];

```

```
if (!eregi("^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,3})$", $email))
{
    echo "Alamat Email anda Salah ...";
}

else
{
    echo "Alamat Email yang anda masukkan adalah $email";
}
?>
```

anda bisa mendalami regex lebih dalam dengan membaca tutorial-tutorial yang saya bawakan di ilmuwebsite.com di dalam file latihan saya menyediakan satu file berisi 15 pola regex yang sering digunakan oleh web developer. dan tentunya menggunakan php.

7.3 Mengupload file

Dalam pembuatan aplikasi berbasis web, mengupload file merupakan sesuatu yang sangat di perlukan dalam php, syntax untuk mengupload file pun sudah di sediakan ... ada sedikit perbedaan antara form yang biasa dengan form yang digunakan untuk mengupload file. Terdapat satu atribut tambahan untuk form tersebut, kita coba akan buat terlebih dahulu form yang digunakan untuk mengupload file nantinya ...

```
<html>
<head>
<title>Form Upload</title>
</head>
<body>
<form enctype="multipart/form-data" action="upload.php" method="post">
<p>
<!-- MAX_FILE_SIZE digunakan untuk membatasi size/ukuran file yang akan
di upload -->
<input type="hidden" name="MAX_FILE_SIZE" value="102400" />

<input type="file" name="file" /><br/>
<input type="submit" value="upload!" />
</p>
</form>
</body>
</html>
```

Kita coba lihat dulu di browser. Oke, selanjutnya kita akan membuat sebuah file untuk menerima inputan dari form upload tersebut untuk selanjutnya di letakkan di server dalam folder yang terpilih. dalam hal ini saya meletakkannya dalam folder upload, namun terlebih dahulu kita buat 1 buah folder dengan nama upload. Setelah beres kita coba buat script untuk menyimpan file yang di upload ke dalam folder upload tersebut :

```
<?php
// ketika filenya dikirim ...
if ( isset( $_FILES['file'] ) ) {

// file yang dikirim memiliki atribut sesuai dengan nama input file
// pada formnya
// tadi saya membuat input file upload dengan nama file
// sehingga dalam mengakses nama file tersebut
// kita cukup memanggil $_FILES['file']['nama']
// seperti ini
```

```

echo "name: ".$_FILES['file']['name']."<br />";
// begitu juga jika ingin mengetahui atribut yang lain seperti sizenya,
// tmp_namenya,
// type data yang di upload...
// dan lain lain sebagainya ..

echo "size: ".$_FILES['file']['size']." bytes<br />";
echo "temp name: ".$_FILES['file']['tmp_name']."<br />";
echo "type: ".$_FILES['file']['type']."<br />";
echo "error: ".$_FILES['file']['error']."<br />";

// lakukan pemilihan file
// disini hanya dapat menerima file dengan tipe image/jpeg
// atau gmabar ...
// jika gambar yang di upload itu bertipe jpeg atau gambar

if ( $_FILES['file']['type'] == "image/jpeg" ) {
// kemudian ambil letak file temporary,
// ini dilakukan karena ketika file itu di upload
// server php dalam hal ini apache secara otomatis meletakkannya
// sebagai temporary file
$source = $_FILES['file']['tmp_name'];

// kemudian buat target atau akan di letakkan di mana file tersebut
$target = "./upload/".$_FILES['file']['name'];

// setelah file temporarinya sudah di definisikan
// dan kemudian file tujuan akhir, atau akan di letakkan di mana file
// yang akan di upload tadi
// selanjutnya adalah memindahkan file temporari tersebut ke dalam
// folder tujuan, yakni folder upload
move_uploaded_file( $source, $target );

// untuk mendapatkan ukuran atau size dari image yang di upload ...
$size = getimagesize( $target );

// kemudian tampilkan image yang telah di upload tersebut ke dalam
// halaman yang saat ini di akses ...
// dalam hal ini file tersebut adalah upload.php
$imageupload = "<p><img width=\"$size[0]\" height=\"$size[1]\""
" . "src=\"$target\" alt=\"uploaded image\" /></p>";

echo $imageupload;
}

```

```
}
```

```
?>
```

Oke, langsung saja kita tes di browser ... dan hasilnya adalah ... file yang di upload akan secara otomatis di kopikan kedalam folder upload, ini bisa anda lihat pada direktori c:\xampp\htdocs\xxx\upload. Oke setelah semua selesai anda bisa coba untuk mengulangnya kembali agar anda semakin terbiasa dan mahir nantinya ...

Selanjutnya saya akan bahas bab 8 mengenai pengiriman email menggunakan php.

Bab 8:

Mengirim Email

menggunakan PHP

8.1 Mulai mengirim email

Ketika seorang user melakukan pendaftaran akun pada suatu website, secara otomatis biasanya website tersebut mengirim email yang memberitahukan jika user tersebut telah terdaftar, dan perlu dilakukan aktivasi sebelum dapat digunakan. Begitu juga ketika pemberitahuan akan adanya news update, seorang admin website secara otomatis mengirimkan berita melalui email secara otomatis.

Fitur email biasanya disediakan oleh hosting yang anda bayar, di localhost pun anda bisa mengirim email dengan fasilitas mercury dari xampp, namun saya tidak akan menjelaskan untuk setting email menggunakan mercury di localhost, karena untuk mengirim email anda memerlukan DNS Server yang memiliki IP Publik tersendiri sehingga dapat mengirim email, dan email yang dikirim pun jika DNS nya tidak terregistrasi maka akan dianggap sebagai bulk email atau spammer, oleh karena di sini saya hanya jelaskan bagaimana mengirim email pada hosting. Oke pertama siapkan terlebih dahulu hostingnya ... untuk percobaan anda dapat menggunakan hosting-hosting yang gratisan. Sebagai contohnya di sini saya akan menggunakan <http://www.byethost.com/> Anda dapat melakukan registrasi terlebih dahulu di situ. Setelah beres, kita coba masuk ke dalam salah satu direktori. Terus kita buat filenya **mail.basic.php**, untuk mengirim email menggunakan php, syntaxnya adalah seperti ini :

```
mail(to,subject,message,header,parameter)
```

sehingga di **mail.php** kita ketika seperti ini, sebagai contoh ...

```
<?php
$to = alkemail@gmail.com
$subject = "Email Contoh";
$message = "Ini adalah email Contoh ... ";
$header = "From: <alk@bytehouse.com>";
// kemudian untuk mengirim email

$sent = mail($to, $subject, $message, $header);

if($sent)
{
    echo "Email contoh sudah terkirim...";
}
else
{
    echo "Gagal kirim email!";
}
?>
```

8.2 Mengirim email dalam bentuk HTML

Antara Mengirim email bentuk biasa, dengan mengirim email dalam bentuk html, tidak terlalu berbeda, yang jadi perbedaan hanya di bagian headernya saja, di dalam email format html anda perlu menambahkan beberapa atribut ke dalam header. Di mana header inilah yang berperan penting untuk menjelaskan kepada email reader itu sebagai email dalam format html. Sebagai contohnya ... kita akan buat satu buah script yang dapat mengirim email menggunakan HTML.

```
<?php
$to = alkemail@gmail.com
$subject = "Email Contoh";
$message = "<h1>Ini adalah email Contoh</h1>... <br />Tutorial Terbaru
Web Development silahkan kunjungi ...
Ilmuwebsite.com";

// Always set content-type when sending HTML email
$header = "MIME-Version: 1.0" . "\r\n";
$header .= "Content-type:text/html;charset=iso-8859-1" . "\r\n";

$header .= 'From: <alk@bytehouse.com>' . "\r\n";
$header .= 'Cc: <myboss@example.com>' . "\r\n";

$sent = mail($to, $subject, $message, $header);

if($sent)
{
    echo "Email contoh sudah terkirim...";
}
else
{
    echo "Gagal kirim email!";
}
?>
```

oke, langsung saja kita coba ...

8.3 Menyisipkan gambar dalam email format html

Untuk menyisipkan gambar dalam email format html, anda cukup memanggil letak gambarnya saja, kemudian di letakkan di bagian src nya ... sama seperti ketika anda membuat halaman html yang memuat gambar. Kita buat script untuk mengirim email ..

```
<?php
$to = alkemail@gmail.com
$subject = "Email Contoh";
$message = "<h1>Ini adalah email Contoh</h1>... <br />Tutorial Terbaru
Web Development silahkan kunjungi ...

```

Bab 9:

Autentikasi dengan htaccess, htpasswd, Session dan Cookies

9.1 Menggunakan Session dan Cookies untuk halaman login

Session dan Cookies boleh dibilang merupakan sesuatu yang sangat penting dalam pembagian hak akses, antara user yang belum login, dengan user yang sudah login... dalam wordpress misalnya, cuma user yang sudah login saja yang bisa memposting artikel, namun user yang belum login tidak bisa melakukannya. Contoh yang lain... misalkan dalam forum, anda tidak diperkenankan membuat thread baru jika anda belum melakukan login, mungkin contoh yang lainnya, anda tidak bisa membuat wall status baru jika anda belum login ke facebook anda.

Semua berhubungan dengan login, namun apa yang berada di balik kode-kode login tersebut, hampir semua login pada aplikasi web atau website itu menggunakan session dan cookies untuk membedakan kondisi user yang sudah login dan belum login, namun sebelum mengetahui lebih lanjut, saya akan coba jelaskan terlebih dahulu apa itu session dan apa itu cookies...

Session

Session merupakan variable global yang jika diregistrasikan atau didefinisikan maka variable tersebut dapat di akses di halaman manapun. ketika kita meregistrasikan session, maka php akan kepada browser untuk menyimpan session yang ada, sehingga

session disimpan oleh browser sementara mirip dengan pengandaian sebuah tiket masuk. setiap halaman php yang membutuhkan adanya login akan mengecek apakah variable sessionnya sudah diregistrasikan. jika sudah maka, halaman php tersebut akan mempersilahkan user dapat mengaksesnya...

Oke, karena session memiliki tugas khusus dan session bertugas sebagai variable global yang dapat di akses di halaman mana saja, sehingga session pun membutuhkan fungsi tersendiri untuk dapat menjalankannya.

Standarnya membuat session adalah seperti ini :

```
<?php
// fungsinya terlebih dahulu
session_start();

// dan standarnya setiap session_start digunakan
// dan ketika user mengakses halaman tersebut biasanya memiliki
// session_id tersendiri kita tes
echo session_id();

// session_id() akan selalu berbeda setiap anda menjalankan browser
?>
```

kita akan buat 2 buah file, file pertama digunakan untuk meregistrasikan/mengeset session, file yang kedua berguna sebagai halaman untuk mengetes apakah sessionnya telah berfungsi dengan benar, jika benar maka file yang kedua itu dapat menampilkan isi dari session yang sebelumnya telah diregistrasikan pada halaman pertama ...

file pertama, **session_reg_first.php** :

```
<?php
// untuk membuat session, diperlukan fungsi khusus yang dapat
// memproduksi session
// yakni, dan jangan lupa untuk meletakkan sessio_start di baris paling
// awal setelah <?php

session_start();

$_SESSION['first'] = "saya adalah session";
echo "Anda telah meregistrasikan session berisi '". $_SESSION['first'] . "</strong>'";
session_destroy();
```

```
?>
<br />
<a href="other_page.php">klik di sini untuk pindah halaman</a>
```

kemudian file selanjutnya **other_page.php** :

```
<?php
session_start();
echo "Anda telah membuat session di halaman sebelumnya, berisi
'<strong>" . $_SESSION['first']."' . '</strong>';
session_destroy();
?>
```

Kita coba tes di browser. Begitu juga jika anda ingin mengakses session yang telah dibuat pada halaman lain. Misalkan kita buat satu buah file sembarang yang file ini juga di coba untuk mengakses. Kita beri nama file tersebut misalnya random.php

```
<?php
// jangan lupa untuk memasukkan fungsi session_start setiap akan
// meregistrasi atau mengakses session
session_start();
// kita coba akses variable session yang telah di set
echo "isi dari session yang aktif ".$_SESSION['first'];
?>
```

Menghapus atau menonaktifkan Session

Ada satu pertanyaan bagaimana caranya untuk menonaktifkan atau menghilangkan session yang telah dibuat :

```
<?php
session_start();
unset($_SESSION['first']);
echo "Isi '".$_SESSION['first']."' adalah ... = " . $_SESSION['first'];
// jika ingin memusnahkan semua session yang ada
// anda dapat menggunakan session_destroy
// biasanya hal ini digunakan ketika proses logout terjadi
// semua session yang ada benar-benar di hapus
// penggunaanya adalah seperti ini
session_destroy();
```

```
echo "<br />Semua session telah di hapus ...";
?>
```

Cookie

Jika session itu akan berakhir ketika browser ditutup, namun jika cookie itu akan berakhir pada waktu yang ditentukan oleh web developer sebelumnya. Contoh penggunaan cookie adalah seperti ini :

```
// pertama kita buat terlebih dulu file cookiel.php
<?php
// lalu di sini kita siapkan variable untuk cookienya
$isicookie = "ini adalah isi dari cookie";
// kemudian kita buat cookienya dengan lama waktu 1 jam misalnya ...
// di hitung dengan menggunakan satuan waktu detik
setcookie("cookiel", $isicookie, time() + 3600);
// untuk dapat mengakses cookie anda dapat menggunakan sintax
echo $_COOKIE["cookiel"];
?>
```

Dan misalkan kita ingin mengakses cookie di halaman yang lain juga bisa, kita buat halaman lain untuk mengakses cookie :

```
<?php
// kemudian langsung akses variable cookienya
echo $_COOKIE["cookiel"];
?>
```

Begitu jika ingin menghapus atau menghilangkan cookie yang ada, kita buat satu buah file lagi **deletecookie.php**

```
<?php
// hapus cookie yang ada dengan unset
unset($_COOKIE['cookiel']);
echo $_COOKIE['cookiel'];
?>
```



Jl. Panduraya No.197 Ruko No.6, Bogor Utara
Telp. [0251] 8335263 | Fax. [0251] 8382444
Email. greentechnomedia@gmail.com
Website. www.greentechnomedia.com



Selayang Pandang
Green Technomedia Advertising

Info Green Technomedia
<http://www.greentechnomedia.com>
<http://www.facebook.com/greentechnomedia>

Informasi Jasa
Galih Priyaji [+62 856 9190 3238]
Risky Irawan [+62 856 7705 163]

Green Technomedia (GTM), merupakan perusahaan yang bergerak di bidang *Advertising*, Teknologi Informasi, dan *Multimedia Art* yang menerapkan pola kemitraan dengan selalu berpijak pada nilai-nilai profesionalisme, pengembangan teknologi, dan peningkatan kualitas sumber daya sebagai landasan dasar dalam memberikan layanan terbaik demi kepuasan pengguna jasa.

ADVERTISING

Graphic Design

Logo, Corporate Identity, print ad, packaging, web design, company profile, catalogue, etc.

Animation & Presentation

Interactive Multimedia, 2D Animation, Motion Graphic, 3D Animation, 3D Modelling, etc.

Videography

Product Promotion, Video Clip Short Movie, Event Documentation Editing & Capture, etc.

Photography

Documentation, Commercial Architecture, Art & Product, etc.

WEB DEVELOPMENT

Personal, Business, Blog, E-Commercial, CMS, SEO, etc.

SOFTWARE DEVELOPMENT :

Accounting, procurement, Purchasing, GIS, Expert System, etc.

Advertising

Dalam setiap iklan yang baik terdapat sebuah konsep kreatif, sebuah gagasan besar yang membuat pesannya berbeda atau unik, dapat mencuri perhatian banyak orang, mudah diingat, dan menjadi penentu ide penjualan utama yang akan menjadi tema kampanye sebuah iklan. Dalam proses membuat iklan yang kreatif dan menarik itu, sangatlah perlu peran dari *Creative Team* dalam menciptakan pesan dan konsep yang menjual karena pengembangan strategi kreatif dituntun oleh tujuan dan sasaran pesan. Kami sebagai mitra dan juga berperan sebagai *Creative Team* dapat membantu menuangkan ide-ide kreatif usaha/bisnis anda kedalam media *Audio*, *Visual* dan *Audio Visual* yang didukung oleh seni *photography* dan *videography* yang menarik untuk menarik konsumen, baik dari sisi *Mind Share (Segmentation, Targeting, Positioning)*, *Market Share (Differentiation, Marketing Mix, etc)* dan juga *Heart Share (Value & Branding)*.

Web Development

Dengan adanya teknologi *internet*, perusahaan dapat semakin terbantu untuk mengenalkan aneka produk dan jasanya kepada target pasar tertentu.

Bukan hanya itu, untuk para profesional, Internet akan mempromosikan diri dan usaha anda 24 Jam sehari. 7 Hari Seminggu. *Non Stop!* Internet bisa diakses oleh siapa saja, dimana saja, dibelahan bumi manapun yang terhubung dengan *Internet*, sehingga bisnis anda akan terlepas dari kendala jarak dan waktu. *Internet* akan mengenalkan anda pada saudara baru, teman baru, ataupun partner bisnis baru. Salah satu fasilitas yang paling efektif pada internet adalah **website**. Karena itu, kami ingin menawarkan kepada Anda tentang pembuatan **website** yang menarik, interaktif dan mudah di akses, disertai dengan fitur *affiliate marketing* yang bisa mendongkrak penjualan produk dan jasa anda.

Software Development

Kami juga dapat memfasilitasi berbagai bidang garap bisnis melalui penyediaan sistem informasi manajemen. Saat ini program manajemen merupakan program yang membantu manajemen puncak dan para staff untuk menangani semua informasi di perusahaan baik itu departemen *accounting*, *purchasing*, *sales*, ataupun membuat laporan-laporan yang dibutuhkan pihak *stakeholder* dalam mengembangkan manajemen perusahaan.

Bab 10:

Studi kasus Sistem Informasi Administrasi Mahasiswa

Setelah kita mendalami bagaimana menggunakan PHP dan MySQL pada bab-bab sebelumnya, tentunya saat ini pengetahuan anda akan PHP dan MySQL jelas berbeda ketika anda belum membaca ebook buatan penulis yang ganteng ini . Setidaknya yang saya harapkan dari pembuatan ebook php ini adalah skill anda sekarang sedikit bertambah dari sebelumnya, jika saat ini masih terdapat kesamaan dengan sebelumnya, berarti saya menyarankan kepada anda untuk melihat ebook php ini dari awal sekali lagi, namun jika anda telah berkali kali melihat ebook php ini dan tidak ada penambahan skill, maka sebaiknya anda mengirimkan pertanyaan ke email saya di alkemail@gmail.com.

Skill dasar hingga menengah untuk php dan mysql telah didapatkan, nah sekarang bagaimana mengimplementasi skill yang telah kita dapatkan sebelumnya. Oke, kita akan membuat sistem informasi Administrasi Mahasiswa.

Namun Sistem Informasi Yang satu ini pun memiliki beberapa fitur layaknya sebuah blog, anda dapat memasukkan berita/artikel, dan user bisa memberikan komentar dari berita/artikel yang anda masukkan.

Sistem Informasi Administrasi ini memiliki fitur antara lain :

1. Form Pendaftaran Mahasiswa (FrontEnd)
2. Daftar Calon Mahasiswa Baru (BackEnd)

3. Form Pengisian Nilai Mahasiswa (BackEnd)
4. Daftar Nilai Mahasiswa (BackEnd)
5. Fitur News/Article Update Ala Blog (Frontend BackEnd)
6. Fitur Comment Article Ala Blog (FrontEnd)

FrontEnd disini berarti, di peruntukkan untuk user/publik saja yang bisa melihat. Sedangkan BackEnd disini berarti, hanya si admin saja yang dapat mengakses, termasuk didalamnya menambah, melihat, mengedit, ataupun menghapus. (CRUD) Create, Read, Update, Delete.

10.1 Merancang Sistem Informasi

Hampir dalam semua bidang, ketika kita akan membuat satu produk, baik itu dari bidang kimia, mesin, dan lain sebagainya, semua di rancang terlebih dahulu, sebelum akhirnya benar-benar di buat. Perancangan ini perlu karena, dalam perancangan inilah semuanya benar-benar diperhitungkan secara matang.

Dalam proses perancangan sistem informasi, yang harus dibuat pertama kalinya adalah, desain dari aplikasi yang akan kita buat. Biasanya saya sendiri mengawalinya dengan pseudo code, artinya rancangan yang berada dalam otak kita, apa yang akan kita buat, dan selanjutnya kita konversikan ke dalam desain interface. Di sini saya hanya akan menjelaskan desain interface nya saja. Berdasarkan fitur-fitur yang kita inginkan, ada beberapa desain interface yang akan kita buat.

Kita bagi berdasarkan peletakan halaman, satu untuk admin dan satu untuk user/public

admin

- Halaman Login Admin
- Halaman Pengisian Nilai Mahasiswa
- Halaman Daftar Nilai Mahasiswa
- Halaman Daftar Mahasiswa
- Halaman Daftar Calon Mahasiswa Baru
- Halaman Pembuatan/Penambahan/Pengeditan Artikel Baru
- Halaman Daftar Artikel
- Halaman User Edit+Delete+Add
- Halaman Edit+Delete Komentar

user/publik

- Halaman Login
- Halaman Registrasi User Baru
- Halaman Pendaftaran Mahasiswa Baru
- Halaman Komentar
- Halaman Melihat Nilai Mahasiswa

10.2 Mempersiapkan database

Table yang harus dipersiapkan dalam pembuatan sistem informasi berbasis web yang telah kita rancang sebelumnya :

```
tbl_cln_mhsiswa
tbl_mhsiswa
tbl_user
tb_user_profile
tbl_nilai_mahasiswa
tbl_artikel
tbl_komentar
```

```
create table tbl_cln_mahasiswa (id_daftar int(5) auto_increment primary key, tanggal_daftar date, nama_pendaftar varchar(75), jns_kelamin varchar(15), status varchar(35), lulusan_sekolah varchar(75), tahun_ajaran varchar(30), tgl_lahir date, pekerjaan varchar(120), alamat varchar(200), kelurahan varchar(45), kecamatan varchar(45), kota varchar(45), provinsi varchar(45), telp varchar(50), email varchar(55), website varchar(75));

create table tbl_mhsiswa (id_mahasiswa int(5) auto_increment primary key, nama_mahasiswa varchar(75), jns_kelamin varchar(15), tgl_lahir date, status varchar(35), jurusan varchar(75), nim int(15), lulusan_sekolah varchar(75), tahun_ajaran varchar(30), pekerjaan varchar(75), alamat varchar(200), kelurahan varchar(45), kecamatan varchar(45), kota varchar(45), provinsi varchar(75), telp varchar(50), email varchar(55), website varchar(75));

create table tb_user_profile (id_user int(5) auto_increment primary key, nama varchar(75), tgl_lahir date, jns_kelamin varchar(15), status varchar(50), pekerjaan varchar(50), alamat varchar(200), kelurahan varchar(75), kecamatan varchar(75), kota varchar(75), provinsi varchar(75), telp varchar(50), email varchar(55), website varchar(75));

create table tbl_nilai_mahasiswa (id_nilai int(5) auto_increment primary key, nim int(20), mata_kuliah varchar(50), nilai_mahasiswa varchar(3), dosen_mata_kuliah varchar(50));

create table tbl_artikel (id_artikel int(5) auto_increment primary key, tanggal_publish date, penulis varchar(50), judul_berita varchar(200), isi_berita text, status varchar(20));

create table tbl_komentar (id_komentar int(5) auto_increment primary key, id_berita_kampus int(5), tanggal_komentar date, status
```

```
varchar(20), nama varchar(75), isi_komentar text, email varchar (50),
website varchar(50));

create table tbl_user (id_user int(5) auto_increment primary key,
username varchar(50), password varchar(128), level varchar(50));
```

Atau bisa juga langsung melalui file **latihan10.sql** yang disertakan dalam **filelatihanbab10**. Itu adalah daftar dari table yang dibutuhkan dalam membuat sistem administrasi mahasiswa sederhana namun bersifat fundamental.

10.3 Mempersiapkan fungsi, variable, file yang dibutuhkan dalam pembuatan Sistem Informasi

Sistem informasi itu membutuhkan banyak fungsi dalam pembuatannya. Fungsi-fungsi inilah yang kemudian menyusun tubuh dari sistem informasi, sebagai contoh misalnya dalam mempublish artikel nanti ada fungsi yang melihat apakah dalam database itu statusnya published, atau draft, ketika published maka artikel pun akan ditampilkan di halaman depan, jika draft, maka tidak tampil. Begitu juga fungsi untuk menginsertkan data kedalam database, sehingga masing-masing fungsi memiliki tugasnya masing-masing.

Sebelum merancang suatu aplikasi, dan sebelum membangun suatu aplikasi atau sistem informasi secara utuh, yang harus dilakukan pertama kali adalah, mempersiapkan bagian-bagian yang dibutuhkan. Mulai dari pembuatan table, kemudian menyusun fungsi yang sesuai dengan tablenya.

Setelah menyusun fungsi barulah kita menyusun codingnya. Kodenya sudah saya sertakan di dalam **file.latihan.php.rar**.

Selamat Mencoba.