

Manipulasi Database MySQL Dengan PHP

Mohammad Safii

karebet_asli@telkom.net

<http://sapitenk.cjb.net>

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Materi artikel :

- Membuat database mysql dengan PhpMyadmin
- View
- Insert
- Edit
- Delete
- Searching
- Membuat Hitcounter dengan file teks

Syarat :

- Mengerti penggunaan tag HTML
- Browser dapat mengenali sintak Php
- PC sudah terinstall MySql

Php seperti kita ketahui ialah bahasa pemrograman berbasis web. Bahasa ini mempunyai kelebihan yaitu kompatibilitasnya dengan berbagai macam jenis database , dukungan dengan berbagai macam jenis sistem operasi. Php lebih cocok dan umum digunakan jika di gabungkan dengan database mysql. Mysql dengan Php seakan-akan dua hal yang tidak dapat dipisahkan. Tentunya untuk dapat menggunakan keduanya

dibutuhkan tingkat kemampuan programming tertentu. Tulisan ini ditujukan bagi anda yang ingin mulai menyentuh dan mengembangkan aplikasi berbasis Php mysql. Syarat yang harus dikuasai terlebih dahulu ialah anda harus memahai benar-benar HTML.

Banyak tutorial di internet, namun penulis yakin bahwa anda pasti bisa html karena HTML itu sebenarnya sangat mudah. Selain anda harus mahir HTML juga persiapkan juga agar browser anda dapat mengenali sintak Php. Penulis sarankan gunakan PhpTriad, atau easyphp. PhpTriad sudah menyertakan Php, MySql, PhpMyAdmin. Untuk penggunaan Phptriad anda dapat membaca artikel <http://www.ilmukomputer.com/populer/abepoetra-phptriad.php>.

Diasumsikan bahwa php sudah dapat berjalan di browser dan juga anda sudah menginstall MySql. Sekarang buat direktori di folder C:\apache\htdocs, buat direktori dengan nama ikc; sehingga direktorinya c:\apache\htdocs\ikc\. Untuk mengaksesnya anda dapat membuka browser dan mengetikkan <http://localhost/ikc/>

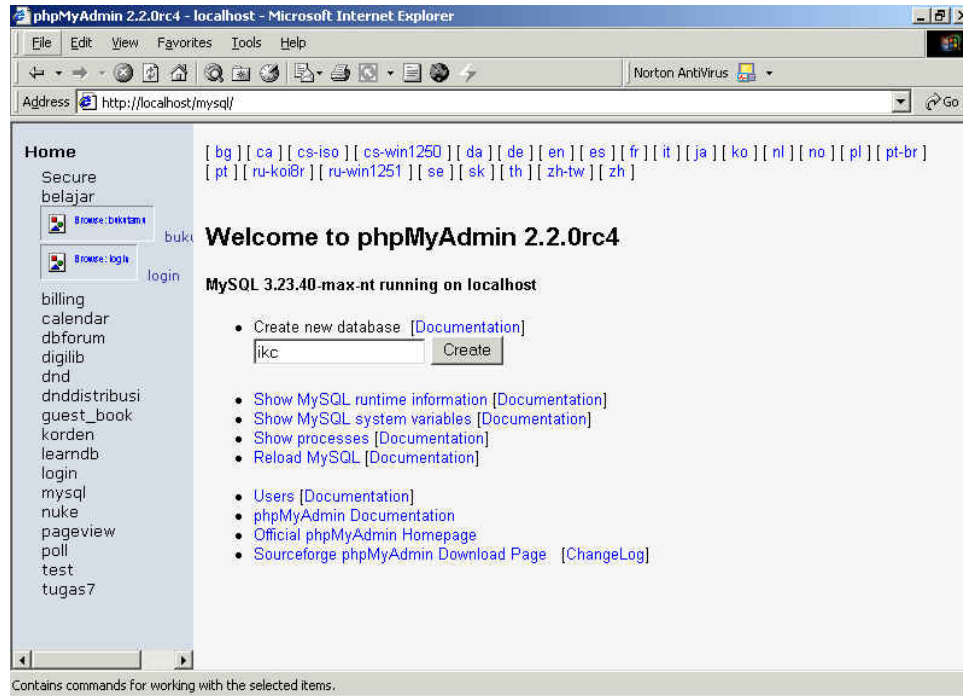
Tutorial ini akan membahas php dan mysql dimana akan dicontohkan bagaimana menghubungkan antara database mysql dengan php, bagaimana menampilkan isi database, update database, hapus database, search database serta berbagai bentuk aplikasi lainnya seperti membuat login dengan php dengan mysql, chat, polling.

Membuat Database mysql dengan PhpMyAdmin

Pertama kita akan membuat database di mysql dan menampilkannya di browser tentunya masih menggunakan Php. Untuk membuat database di mysql sebaiknya gunakan PhpMyadmin. Jika anda menggunakan phptriad untuk mengakses phpmyadmin, pada browser ketikkan <http://localhost/phpmyadmin/>.

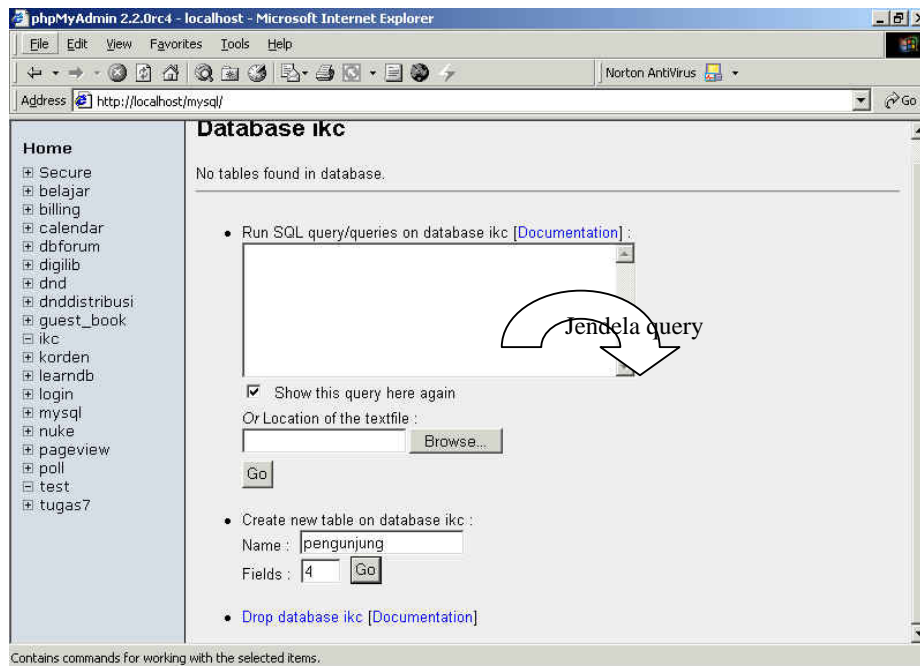
Penulis menggunakan phpMyAdmin 2.2.0rc4. Penulis tidak menggunakan phptriad tapi easyphp (<http://www.easyphp.org>) jadi untuk masalah url di localhost dapat disesuaikan.

Pertama kali buat database mysql dengan phpmyadmin. Beri nama databasenya **ikc**.



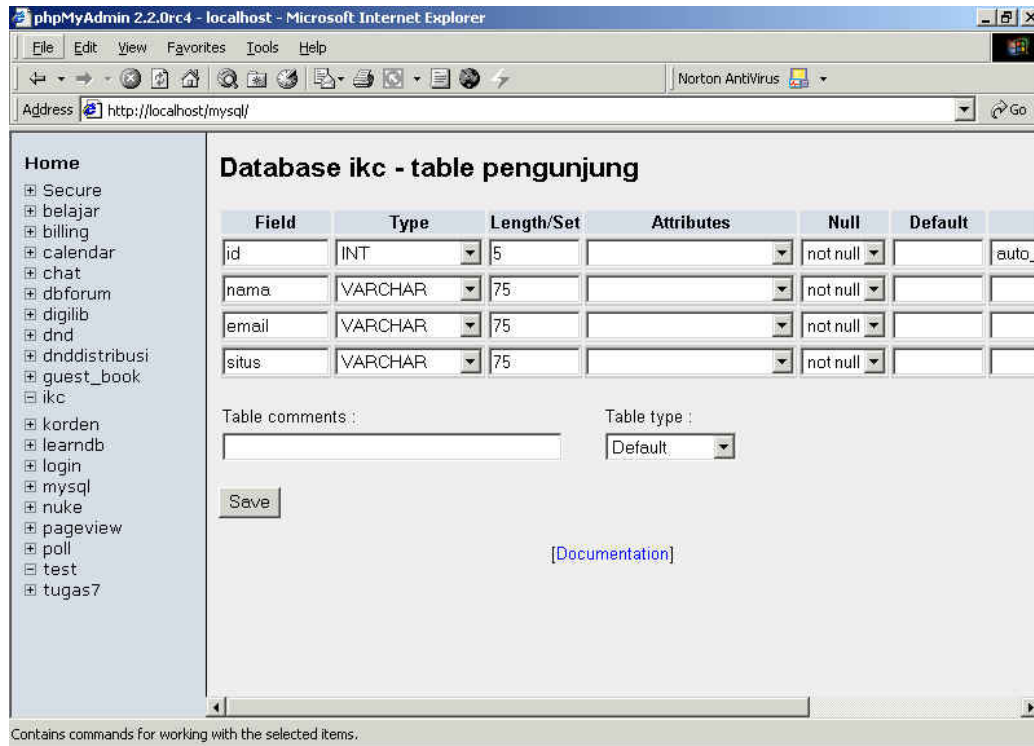
Gambar 1: Database baru

Setelah itu klik **create**. Database baru sudah terbentuk, sekarang kita tinggal membuat tabelnya. Contoh pertama kita buat tabel sederhana yaitu tabel pengunjung pada sebuah halaman web.



Gambar 2: Tabel Pengunjung

Karena kita akan membuat tabel dengan nama **pengunjung** yang mempunyai field sejumlah 4 yaitu : id, nama, email dan situs. Maka masukkan **pengunjung** pada dan 4 pada **fields**. Klik **Go**.



Gambar 3: Membuat field pd database

Selanjutnya ialah mengisi field pada database. Field pertama masukkan **id**,

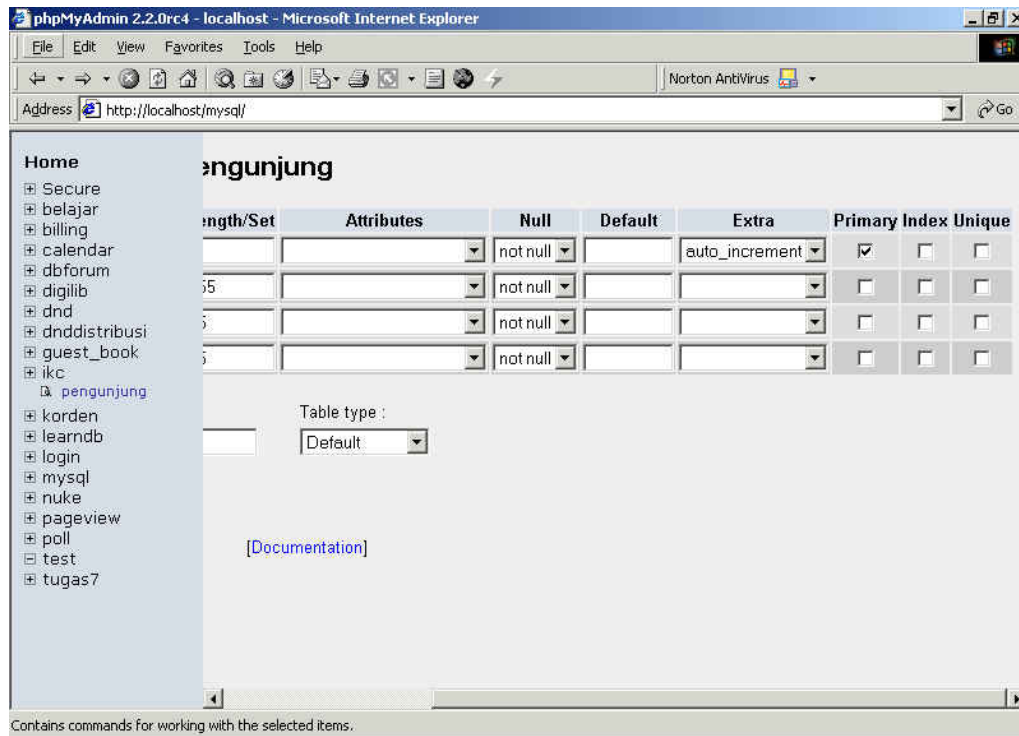
Tabel 1: Field pada tabel pengunjung

Field	Type	Length	Attirbut	Null	Default	extra	P I U
id	Int	5		Not null		Auto_increment	P
nama	Varchar	75		Not null			
email	Varchar	75		Not null			
situs	Varchar	75		Not null			

P : Primary

I : Index

U : Unique

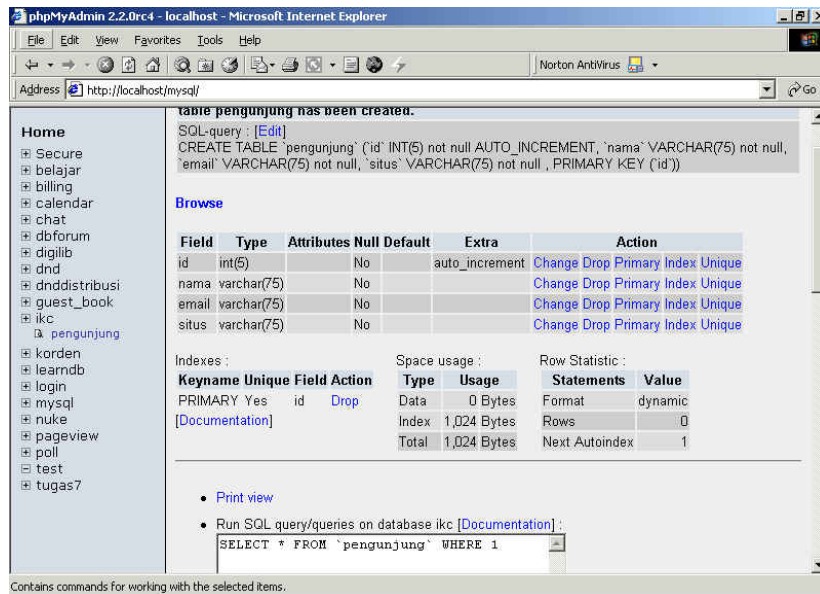


Gambar 4: id di set auto_increement dan primary

Jangan lupa bahwa php *case sensitif* jadi sebaiknya isikan semua field dengan huruf kecil termasuk **id**. Pada gambar 4 **id** di setting auto_increement dan Primary. Auto_increement artinya setiap pengisian record (data) otomatis nilai id akan di naikkan . Primary artinya memberi kunci yang unik yang dapat di jadikan pengenalan dalam tabel (dalam hal ini tabel pengunjung). Angka lebih baik di jadikan kunci utama dari sebuah tabel. Karena angka itu unik dan penggunaannya pun bisa di urutkan sehingga tidak akan ada duplikasi data. Dengan kata lain hanya dengan **id** kita nantinya akan dapat melakukan editing, deleteing, update pada record.

Jika sudah klik **save**.

Nantinya hasil tabel pengunjung seperti gambar 5 :



Gambar 5: Tabel pengunjung

Browse

Insert

Drop

Primary, Index, Unique

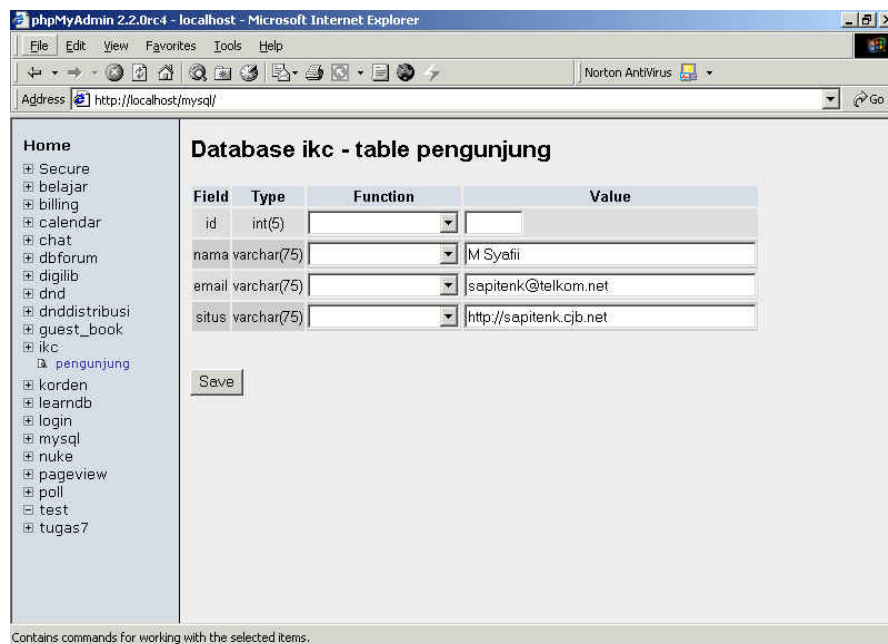
untuk melihat isi table (record)

untuk mengisi table

Menghapus field

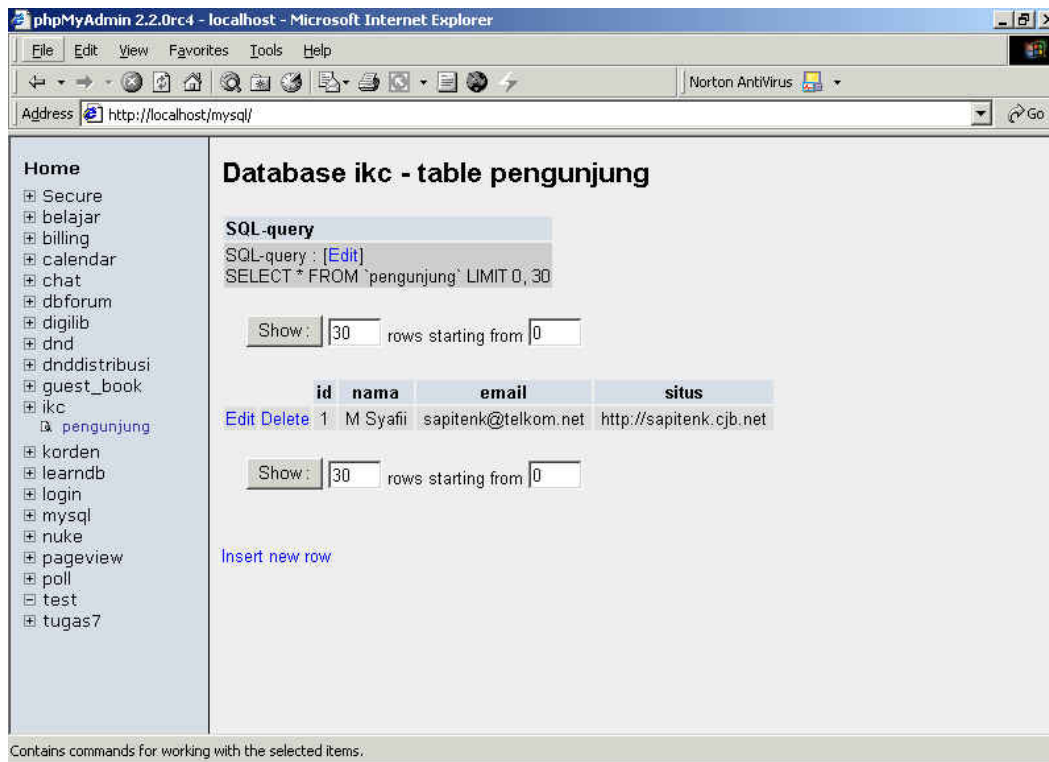
Menjadikan primary, index atau unique

Isi database tersebut dengan beberapa records, klik **Insert** untuk mengisinya :



Gambar 6: Isi Tabel

Karena id **auto_increment** maka id tidak perlu diisi. Jika sudah klik Save. Untuk melihat record klik Browse :



Gambar 7: Browse tabel

Apabila anda tidak menggunakan PhpMyadmin untuk administrasi database MySql maka script yang dapat anda ketikkan di mysql atau paste di jendela query :

```
CREATE TABLE `pengunjung` (`id` INT(5) not null AUTO_INCREMENT, `nama` VARCHAR(75) not null, `email` VARCHAR(75) not null, `situs` VARCHAR(75) not null , PRIMARY KEY (`id`))
```

View

Database sudah terbentuk, sekarang kita akan mulai bagaimana caranya menampilkan isi tabel tersebut pada browser yang tentunya menggunakan Php. Pertama jangan lupa start dulu apache dan mysql anda (jika anda menggunakan PhpTriad).

Tabel 2: Source code view.php

view.php	
1	<?
2	
3	\$host = localhost; // alamat atau host
4	\$user = root; // username
5	\$pass = ""; // password
6	
7	\$koneksi = mysql_connect("\$host","user",\$pass);
8	
9	mysql_select_db("ikc",\$koneksi);
10	
11	\$query = mysql_query("select * from pengunjung",\$koneksi);
12	
13	\$jumlah = mysql_num_rows(\$query);
14	
15	echo "<center>Daftar Pengunjung</center>";
16	echo "Jumlah pengunjung : \$jumlah";
	while(\$baris
17	=mysql_fetch_array(\$query))
18	{
19	echo " ";
20	echo \$baris[0];
21	echo " ";
22	echo "Nama : ";
23	echo \$baris[1];
24	echo " ";
25	echo "Email : ";
26	echo \$baris[2];
27	echo " ";
	echo "Homepage :
28	";
29	echo \$baris[3];
30	}
31	
32	?>

Penjelasan :

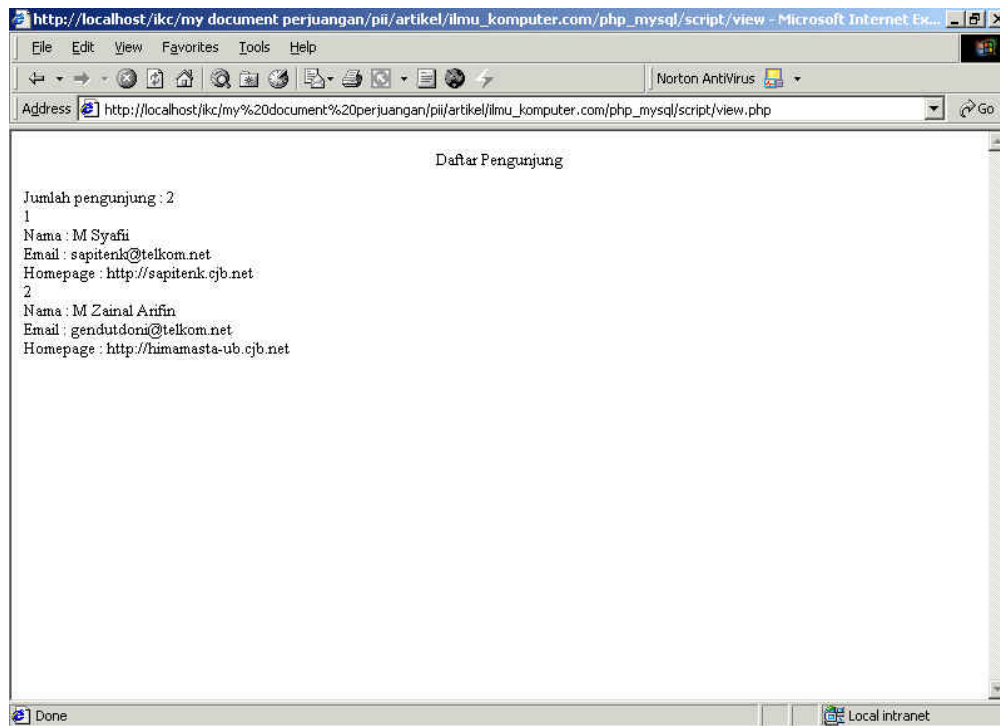
1. Php selalu diawali oleh tag <? Dan diakhiri tag ?>. Seperti pada baris ke 1 dan baris ke 32.
2. Baris ke 7, merupakan cara untuk melakukan koneksi ke mysql. Syarat koneksi ke Mysql ialah memasukkan nama host / IP, username dan tentunya password. Secara default koneksi ke mysql seperti script diatas (baris 7) dengan host = localhost, user = root, dan password di kosongi.
Ringkasnya sintak baris 7 dapat ditulis :

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


3. Setelah terhubung dengan mysql langkah selanjutnya ialah memilih database yang akan digunakan. Pada contoh ini database yang digunakan bernama **ikc**. Baris 9 merupakan sintak bagaimana memilih database.
4. Setelah memilih database **ikc**, maka kita dapat melakukan query. Dengan sintak seperti baris 11 maka kita akan mengambil semua record yang terdapat pada tabel **pengunjung**.
5. Setelah melakukan query maka kita perlu juga menghitung banyaknya record yang terdapat pada tabel. Baris 13 adalah cara untuk menghitung seluruh isi tabel.
6. Langkah utama ialah bagaimana menampilkan isi tabel ke dalam browser. Baris 17 memasukkan semua isi hasil dari query ke dalam variable \$baris. Di dalam \$baris inilah semua isi tabel (record) di tampung. Pada contoh diatas untuk menampilkan setiap urutannya menggunakan urutan dari field yang ada di tabel.

\$baris[0] artinya menampilkan isi field **id**. Cara ini sama dengan \$baris["id"]. Dan seterusnya. Urutan pertama dimulai dengan angka 0 (nol).

Di browser hasilnya :



Gambar 8: View.php

Kesimpulan :

Untuk menampilkan database mysql di Browser dengan Php langkah-langkahnya :

- Hubungkan dulu Php dengan mysql (menggunakan fungsi `mysql_connect()`)
- Pilih database (menggunakan fungsi `mysql_select_db`)
- Query (menggunakan fungsi `mysql_query()`)
- Tampilkan (menggunakan fungsi `mysql_fetch_array`)

Insert

File `view.php` tidak memungkinkan untuk seseorang memasukkan record dikarenakan file tersebut hanya menampilkan record. Untuk dapat membuat sebuah halaman yang dapat memasukkan sebuah inputan maka perlu dibentuk sebuah *form*. Form tersebut memuat variabel-variabel yang dibutuhkan yang kemudian setelah di *submit* akan di proses dan dimasukkan dalam database. Maka dibutuhkanlah dua file yaitu `input.htm` dan file pemroses `input.php`

Tabel 3: Source code input.htm

input.htm	
1	
2	<html>
3	<head></head>
4	<body>
5	<form method="post" action="input.php">
6	<pre>
7	Nama : <input type="text" name="nama">
8	Email: <input type="text" name="email">
9	Situs: <input type="text" name="situs">
10	<input type="submit" name="submit" value="kirim"><input type="reset"
11	name="reset" value="reset">
12	</pre>
13	</form>
	</body>
	</html>

<input type="text" **name="nama"**>

name="nama" artinya bahwa apapun yang di inputkan dalam text box tersebut menjadi variable nama.



Gambar 9: input.htm

Tabel 4: Source code input.php

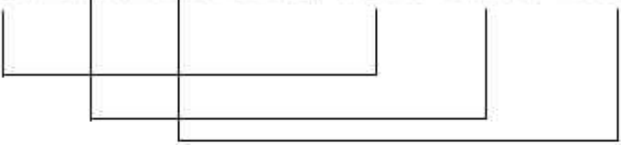
input.php	
1	
2	
3	
4	<?
5	\$koneksi = mysql_connect("localhost","root","");
6	mysql_select_db("ikc", \$koneksi);
7	mysql_query("insert into pengunjung (nama, email, situs)
8	values ('\$nama', '\$email', '\$situs')",\$koneksi);
9	echo "Data telah dimasukkan";
10	?>

Pada source code input.htm terdapat 3 variabel yaitu nama, email dan situs. Ketiga variable tersebut nantinya akan diproses dalam file input.php. Cara pemrosesan atau pemasukan data dalam database mysql ialah menggunakan perintah mysql yaitu perintah **insert**.

**mysql_query("insert into pengunjung (nama, email, situs)
values ('\$nama', '\$email', '\$situs')",\$koneksi);**

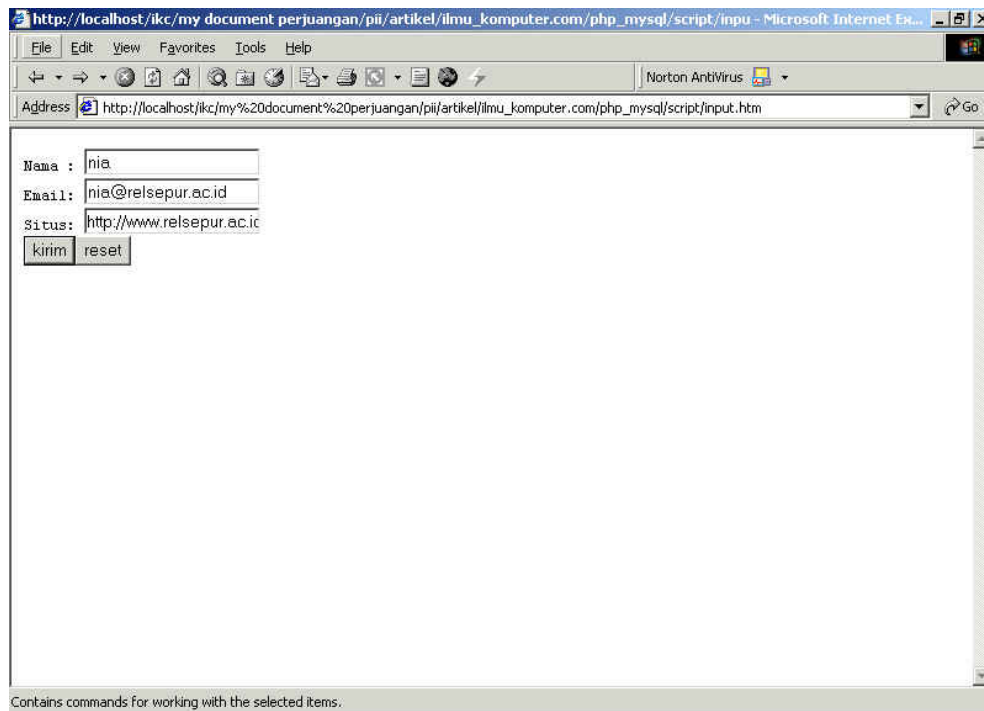
Sintak tersebut memasukkan data yang terdapat pada variable nama, variable email, variable situs untuk di masukkan ke dalam tabel pengunjung dengan urutan field nya nama, email dan situs. **Id** tidak dimasukkan karena sudah **auto_increment**. Memasukkan dalam database harus urut dengan fieldnya. Pada tabel pengunjung urutannya id, nama, email, situs. Maka pada pengisiannya harus di urutkan.

```
insert into pengunjung (nama, email, situs) values ('$nama', '$email', '$situs');
```



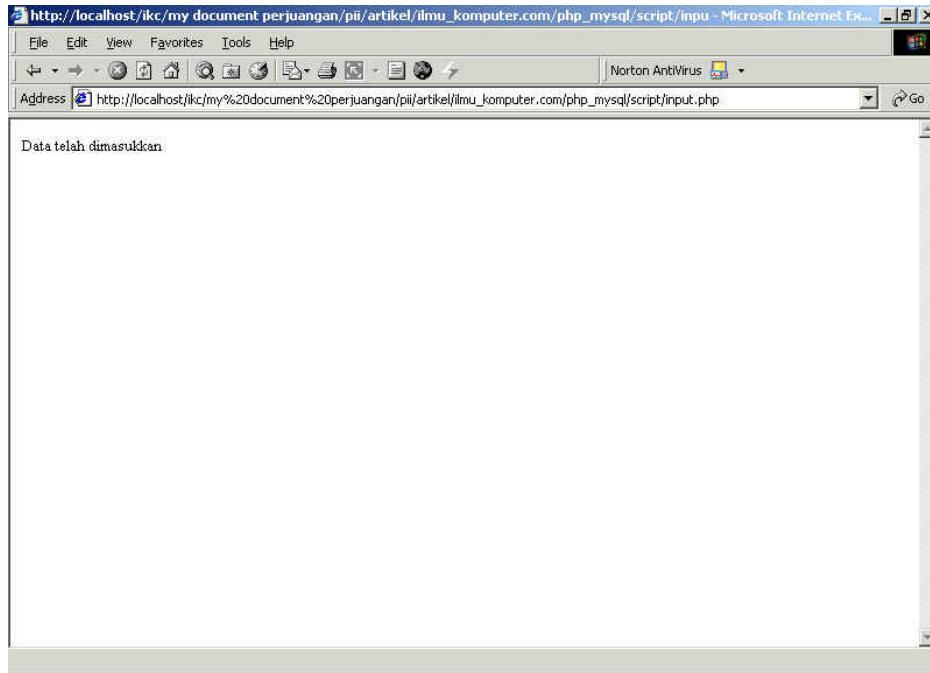
Gambar 10: Urutan pemasukan data

Pengunjung (nama, email, situs) = nama field dalam tabel pengunjung.
Sekarang akan kita coba memasukkan data. Buka file input.htm di browser :



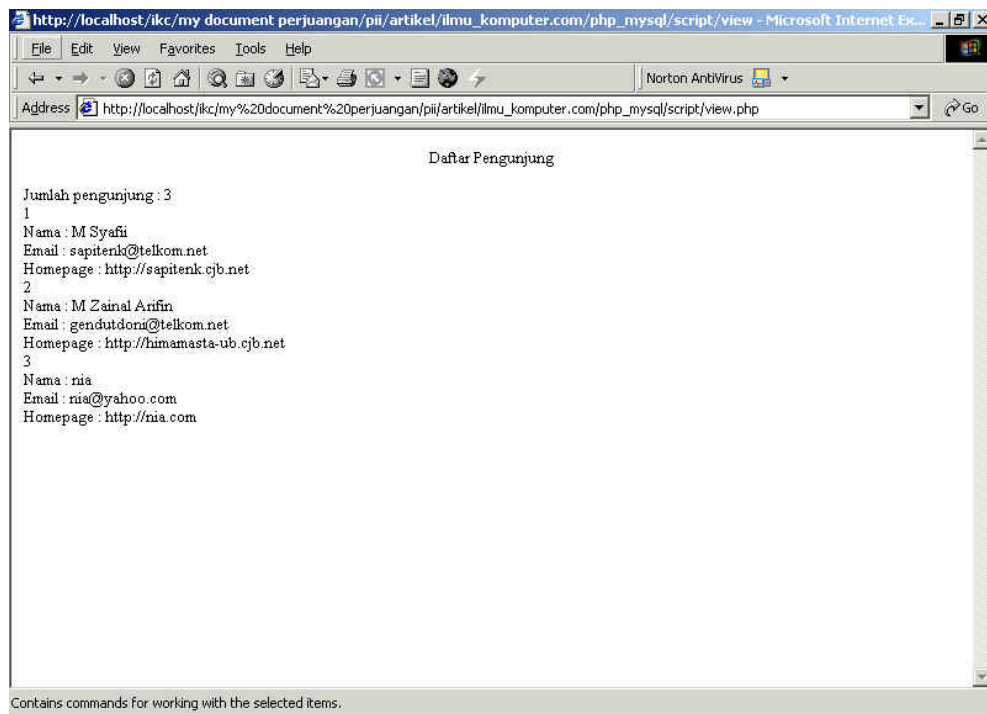
Gambar 11: Input data

Klik kirim, maka selanjutnya akan diproses di input.php



Gambar 12: Input.php

Untuk melihat data buka lagi file view.php



Gambar 13: Tambahan record

Bisa anda lihat bahwa sekarang jumlah record ada 3. **id** yang tidak kita isikan dan masukkan dalam input.php maka di isi otomatis dan nilainya di naikan satu (+1) menjadi 3. Inilah yang dinamakan *auto_increment*. Dari sini anda bisa membuat sebuah form untuk input data. Program diatas juga dapat anda tambahkan yaitu : jika ada nama atau email atau situs yang dikosongi maka data tidak dapat di masukkan ke dalam database karena kosong.

Modifikasi input.php menjadi :

Tabel 5: Source code input.php

input.php	
1	<?
	if((\$nama!="") or (\$email!="") or
2	(\$situs!=""))
3	{
4	\$koneksi = mysql_connect("localhost","root","");
5	mysql_select_db("ikc", \$koneksi);
6	mysql_query("insert into
7	pengunjung (nama, email, situs)
8	values ('\$nama', '\$email', '\$situs')",\$koneksi);
	echo "Data telah
9	dimasukkan";
10	}
11	else
	echo "Cek kembali form
12	anda";
13	?>

Baris 2 akan menyeleksi jika variable nama, email, situs tidak kosong maka masukkan data (jalankan baris 4-9) namun jika kosong maka jalankan baris 11.

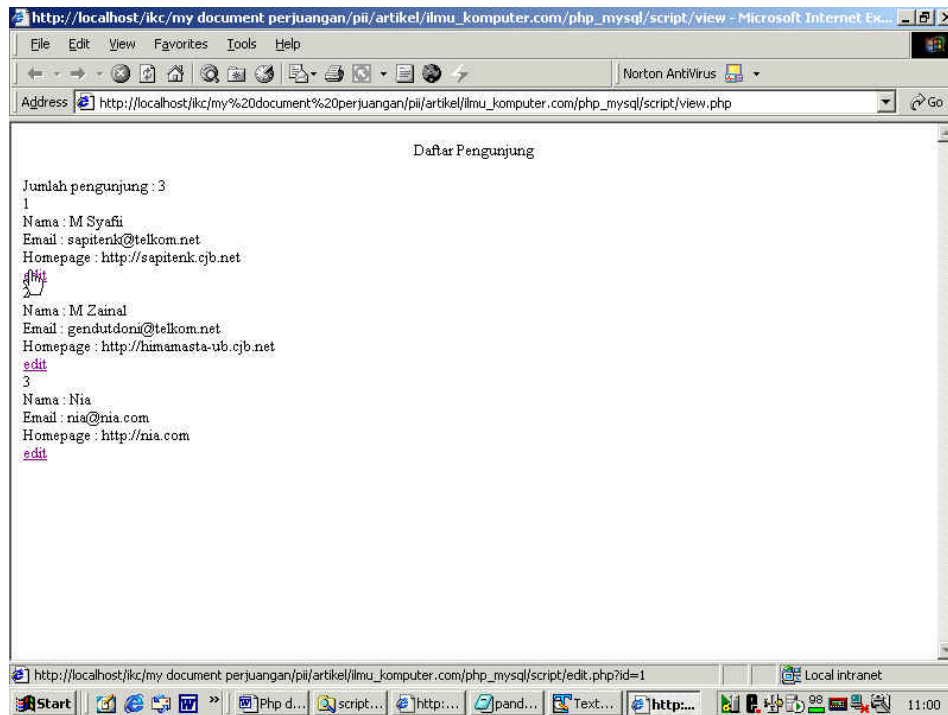
Edit

Sebuah database tentunya terdapat revisi-revisi atau perubahan isi tabel atau record. Maka harusnya terdapat sebuah form yang mampu mengedit isi yang terdapat dalam database. Jika pada file input.php kita sudah dapat memasukkan data ke dalam database maka kali ini kita akan mengedit data yang telah di masukkan. Langkah pertama kita harus memodifikasi file view.php agar dapat mempunyai *link* untuk menuju sebuah form peng-editan yang selanjutnya dengan form tersebut akan di proses dan di update ke dalam database.

Tabel 6: Source code baru view.php

view.php	
1	<?
2	
3	
4	\$koneksi = mysql_connect("localhost","root","");
5	
6	mysql_select_db("ikc",\$koneksi);
7	
8	\$query = mysql_query("select * from pengunjung",\$koneksi);
9	
10	\$jumlah = mysql_num_rows(\$query);
11	
12	echo "<center>Daftar Pengunjung</center>";
13	echo " ";
14	echo "Jumlah pengunjung : \$jumlah";
15	while(\$baris =
16	mysql_fetch_array(\$query))
17	{
18	echo " ";
19	echo \$baris[0];
20	echo " ";
21	echo "Nama : ";
22	echo \$baris[1];
23	echo " ";
24	echo "Email : ";
25	echo \$baris[2];
26	echo " ";
27	echo "Homepage :";
28	echo \$baris[3];
29	echo " edit";
30	}
31	?>

Script view.php mendapat tambahan pada baris 28. Dimana dibuat sebuah link yang memanfaatkan variable **id** untuk di proses pada tahap selanjutnya.



Gambar 14: View.php

Jika cursor diarahkan pada link **edit** maka link tersebut akan menggunakan **id** dari record untuk dijadikan sebagai variable pemroses di file edit.php.

http://localhost/ikc/my%20document%20perjuangan/pii/artikel/ilmu_komputer.com/php_mysql/script/edit.php?id=1

Ketika cursor diarahkan pada record pertama maka link mempunyai nilai id=1. Nilai 1 diambil dari hasil query dari \$baris[0]. Begitu seterusnya. Sehingga dengan mengambil nilai dari **id** maka pada file edit.php nilai **id** tersebut akan di jadikan alat bantu untuk editing. Kenapa harus menggunakan **id** ? karena **id** telah di set primary yang artinya tidak ada *redundancy id*. File edit.php membawa sebuah variable **id**. Dengan variable ini akan di lakukan query yaitu memilih semua field yang mempunyai id = 1 (nilai 1 diambil dari edit.php?id=1); dalam hal ini di contohkan id=1.

Tabel 7: Source code edit.php

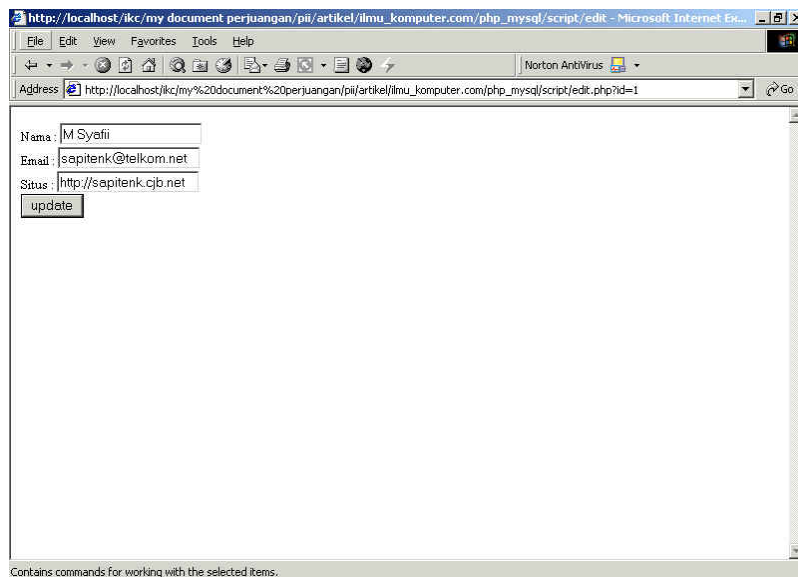
edit.php	
1	<?
2	
3	\$koneksi = mysql_connect("localhost","root","");
4	
5	mysql_select_db("ikc",\$koneksi);
6	\$query = mysql_query("select * from pengunjung where id='\$id'",\$koneksi);


```
while ($baris =  
7  mysql_fetch_array($query))  
8  {  
9  
    echo "<form method=\"post\"  
10  action=\"update.php\">";  
11  echo "Nama : <input type=\"text\" name=\"nama\" value=\"$baris[1]\">";  
12  echo "<br>";  
13  echo "Email : <input type=\"text\" name=\"email\" value=\"$baris[2]\">";  
14  echo "<br>";  
15  echo "Situs : <input type=\"text\" name=\"situs\" value=\"$baris[3]\">";  
16  echo "<br>";  
17  echo "<input type=\"submit\" name=\"submit\" value=\"update\">";  
18  echo "<input type=\"hidden\" name=\"id\" value=\"$baris[0]\">";  
19  echo "</form>";  
20  
21  }  
22  
23  ?>
```

Penjelasan :

Inti dari edit.php yaitu dia membawa sebuah variable id yang nilainya diambil dari view.php. Variable tersebut ditampung dalam variable **\$id** diambil dari edit.php?id=1. Pada baris 6 dilakukan query untuk mengambil seluruh record yang id-nya sama dengan \$id (dalam contoh id=1) .

Baris 18 digunakan sebagai pengendali proses agar jika form diatas di *submit* maka variable **id** tetap dapat digunakan pada file update.php atau jika ingin sebuah variable dapat di ‘tangkap’ untuk diproses maka ketikkan perintah seperti pada baris 18 dan yang penting harus tetap di antara tag <form> dan </form>



Gambar 15: edit.php?id=1

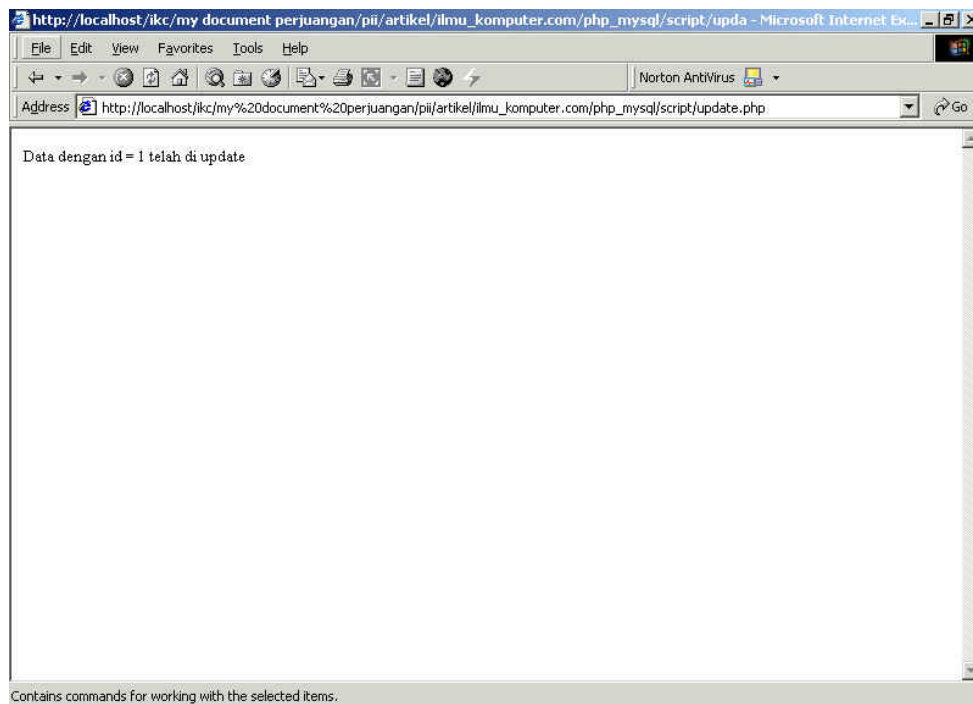
Bisa anda lihat bahwa dengan modal variable **id** saja record tersebut dapat di edit. Logika dari edit.php ialah hasil query di tampung dalam text box. Sehingga nilai (value) yang ada dalam text box hasil query. Silahkan merubah nya jika sudah klik **update**.

Tabel 8: Source code update.php

update.php	
1	<?
2	
3	\$koneksi = mysql_connect("localhost","root","");
4	
5	mysql_select_db("ikc",\$koneksi);
6	\$query = mysql_query("update pengunjung set id='\$id',
7	nama='\$nama', email='\$email', situs = '\$situs' where id='\$id'");
8	echo "Data dengan id = \$id telah di update";
9	?>

Penjelasan :

Baris 6 mengupdate record dimana **id** yang ingin diupdate sama dengan variable **id** yang di 'bawa' oleh file update.php. Jika pada file edit.php baris 18 di hapus maka query tidak dapat dilaksanakan karena nilai variable **id** tidak dapat di akses.



Gambar 16: Update.php

Proses update telah selesai. Anda bisa meliha isi data pada file view.php.

Delete

Record tidak selamanya akan disimpan jika database yang digunakan bersifat dinamis. Dimana setiap waktu atau detik database akan di perbaharui dan data-data lama juga nantinya akan di hapus jikalau sudah tidak sesuai dengan fakta yang ada. Memang seharusnya data-data lama menjadi arsip tapi data-data lama yang ternyata sudah tidak sesuai dengan kebijakan yang baru inilah yang akan di hapus.

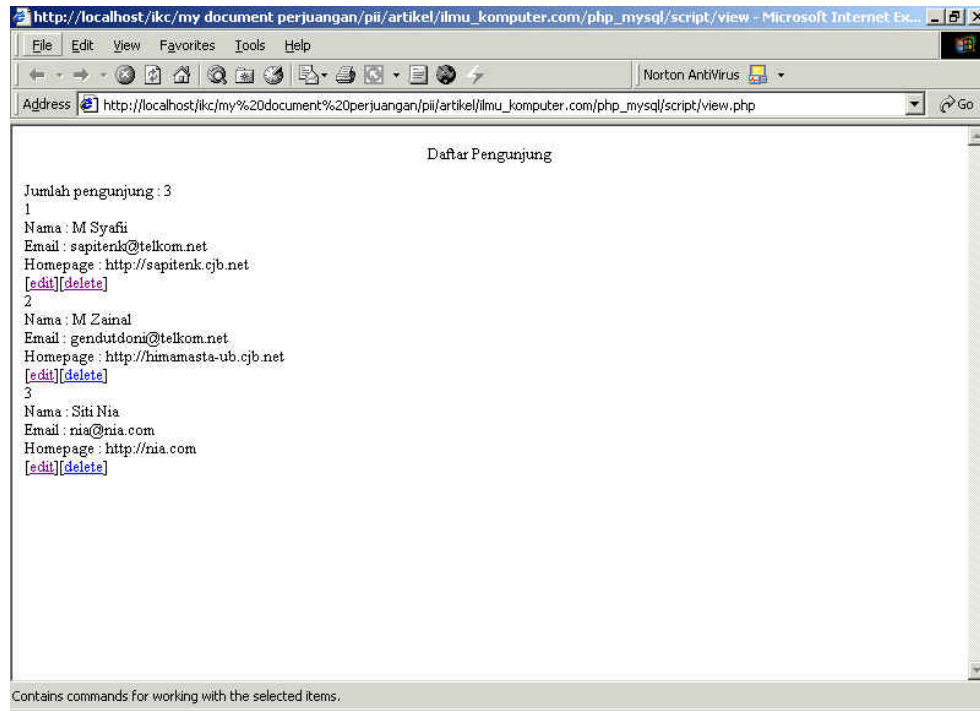
Menghapus database MySql dengan php tentunya sangat mudah. Logika bekerjanya sama dengan edit.php dimana mengambil variable **id** dan di proses di file delete.php. Perintah yang digunakan untuk menghapus adalah perintah MySql yaitu **delete**. File view.php harus kita modifikasi terlebih dahulu.

Tabel 9: Penambahan delete

view.php
<pre>1 <? 2 3 4 \$koneksi = mysql_connect("localhost","root",""); 5 6 mysql_select_db("ikc",\$koneksi); 7 8 \$query = mysql_query("select * from pengunjung",\$koneksi); 9 10 \$jumlah = mysql_num_rows(\$query); 11 12 echo "<center>Daftar Pengunjung</center>"; 13 echo "
"; 14 echo "Jumlah pengunjung : \$jumlah"; 15 while(\$baris = 16 mysql_fetch_array(\$query)) 17 { 18 echo "
"; 19 echo \$baris[0]; 20 echo "
"; 21 echo "Nama : "; 22 echo \$baris[1]; 23 echo "
"; 24 echo "Email : "; 25 echo \$baris[2]; 26 echo "
"; 27 echo "Homepage : "; 28 echo \$baris[3]; 29 echo "
[edit]"; 30 echo "[delete]"; 31 } 32 ?></pre>

Penjelasan :

Untuk mengambil variable **id** ditambahkan perintah seperti pada baris 29. Variable **id** akan di proses pada file delete.php



Gambar 17: view.php dengan delete

Tabel 10: Source code delete.php

delete.php	
1	<?
2	
3	\$koneksi = mysql_connect("localhost","root","");
4	
5	mysql_select_db("ikc",\$koneksi);
6	mysql_query("delete from pengunjung where id='\$id',
7	\$koneksi);
8	echo "Data dengan id = \$id telah di
9	hapus";
10	?>

Kiranya tidak perlu di jelaskan lagi, cara kerjanya sama dengan update.php.

Searching

Database yang besar yang tentunya mempunyai record ribuan atau bahkan ratusan ribu yang seharusnya mempunyai fasilitas pencarian data atau searching. Pencarian ini

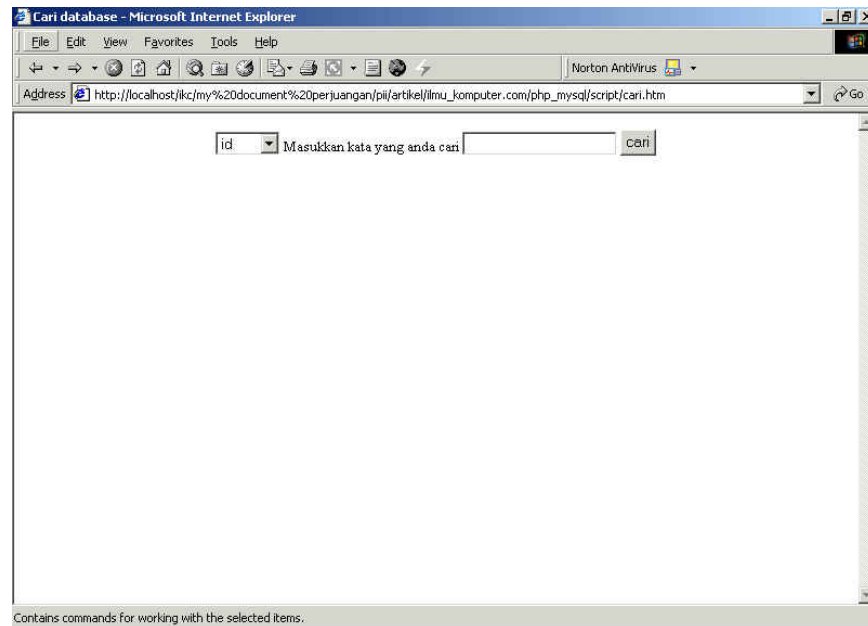
bersifat fleksibel dimana dapat mencari berdasarkan field yang ada dalam tabel. Pada contoh tabel pengunjung, pencarian dapat di *customize*-kan berdasarkan id, nama, email dan situs. Untuk membuat form pencarian menggunakan html sedangkan pemrosesan variable menggunakan php.

Tabel 11: Source code cari.htm

cari.htm	
1	<html>
2	<head><title>Cari database</title>
3	</head>
4	<body>
5	<center>
6	<form name="methode" method="post" action="cari.php">
7	<select name="kolom">
8	<option value="id">id</option>
9	<option value="nama">nama</option>
10	<option value="email">email</option>
11	<option value="situs">situs</option>
12	</select>
13	Masukkan kata yang anda cari <input type="text" type="text" name="cari">
14	<input type="submit" value="cari" >
15	</form>
16	</center>
17	</body>
18	</html>

Penjelasan :

- Baris 8-11 menyatakan variable untuk masing-masing field yang terdapat dalam tabel pengunjung.
- Baris 13 menyatakan kata / keyword yang di cari.



Gambar 18: Cari.htm

Tabel 12: Source code cari.php

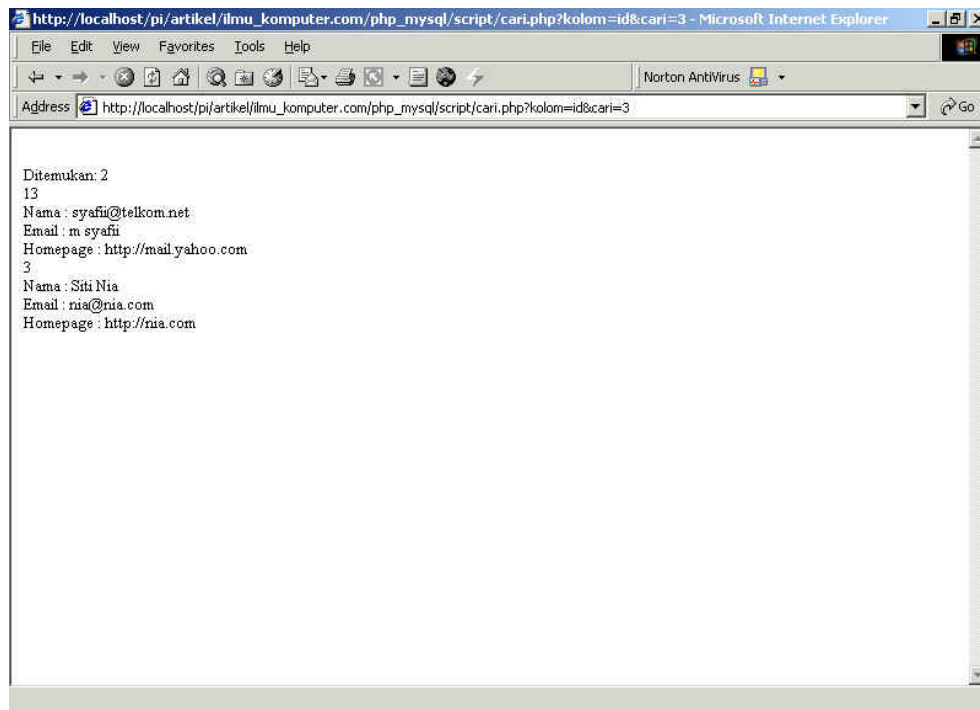
cari.php

```
1  <?
2
3  $koneksi = mysql_connect("localhost","root","");
4
5  mysql_select_db("ikc",$koneksi);
6
7  $query = mysql_query("select * from pengunjung where $kolom like '%" . $cari . "%'
8  ", $koneksi);
9
10 $jumlah = mysql_num_rows($query);
11
12 echo "<br>";
13 echo "Ditemukan: $jumlah";
14 while($baris = mysql_fetch_array($query))
15 {
16 echo "<br>";
17 echo $baris[0];
18 echo "<br>";
19 echo "Nama : ";
20 echo $baris[1];
21 echo "<br>";
22 echo "Email : ";
23 echo $baris[2];
```

```
24 echo "<br>";  
    echo "Homepage :  
25 ";  
26 echo $baris[3];  
27  
28 }  
29  
30 ?>
```

Penjelasan :

- Inti dari script.php ialah pada baris 7. Secara umum query akan melakukan pencarian kata (\$cari) dalam sebuah field (\$kolom) jika cocok akan ditampilkan. Penggunaan like di maksudkan untuk mengambil kata / keyword yang terdapat dalam field yang sepenggal katanya atau huruf sama dengan keyword yang di masukkan. Contoh mencari pengunjung dengan nama **syafii** maka jika menggunakan like akan di tampilkan data **M Syafii, Syafii Maarif**, dll. Sepanjangan kata atau kalimat cocok dengan keyword meskipun satu huruf .



Gambar 19: Hasil pencarian keyword 3

Pada contoh diatas mencari pada kolom (field) **id** dengan keyword **3**. Hasilnya ditampilkan id : **13, 3** .

Membuat Hit Counter

- **Menggunakan file**

Melihat sebuah situs itu bagus atau tidak bisa juga di ukur dari banyaknya pengunjung. Ukuran ramai sebuah situs di karenakan karena design situs yang menarik dan juga materi situs yang sangat inovatif dan selalu update. Nah oleh karena itu diperlukan hit counter untuk melihat statistik sebuah situs. Hit counter ini dapat berupa file ataupun database mysql. Pertama kita akan membuat hitcounter dengan menggunakan file teks.

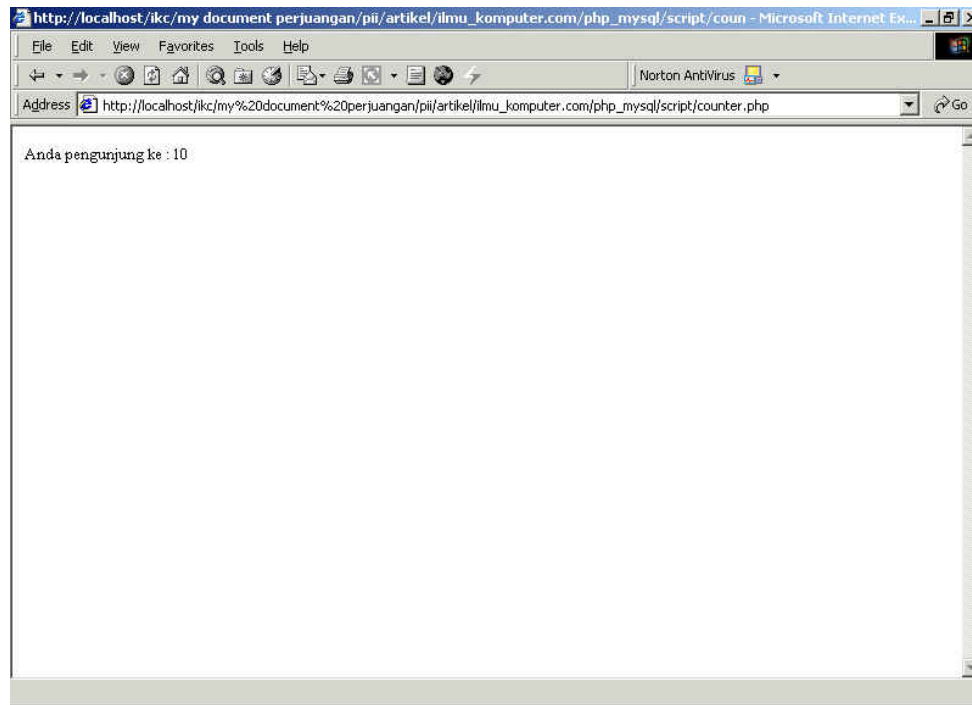
Cara kerjanya ialah setiap pengunjung yang membuka situs kita, nilai counter akan di naikkan satu tentunya sudah di inisialisasi bahwa nilai awal ialah 0 (nol). Berikut source code dari counter.php

Tabel 13: counter.php

counter.php	
1	<?
2	\$log="counter.txt";
3	\$open=fopen(\$log,'r');
4	\$counter=fread(\$open,filesize(\$log));
5	fclose(\$open);
6	\$counter++;
7	\$write=fopen(\$log,'w');
8	fputs(\$write,\$counter);
9	fclose(\$write);
	echo "Anda pengunjung ke :
10	\$counter";
11	?>

Penjelasan :

- Buat dulu file counter.txt
- Baris 3 membuka file counter.txt untuk di baca dan di rubah.
- Baris 6 menaikkan counter yang kemudian di simpan dalam counter.txt.
- Baris 7-8 menulisnya ke dalam file counter.php yang otomatis di naikkan 1.



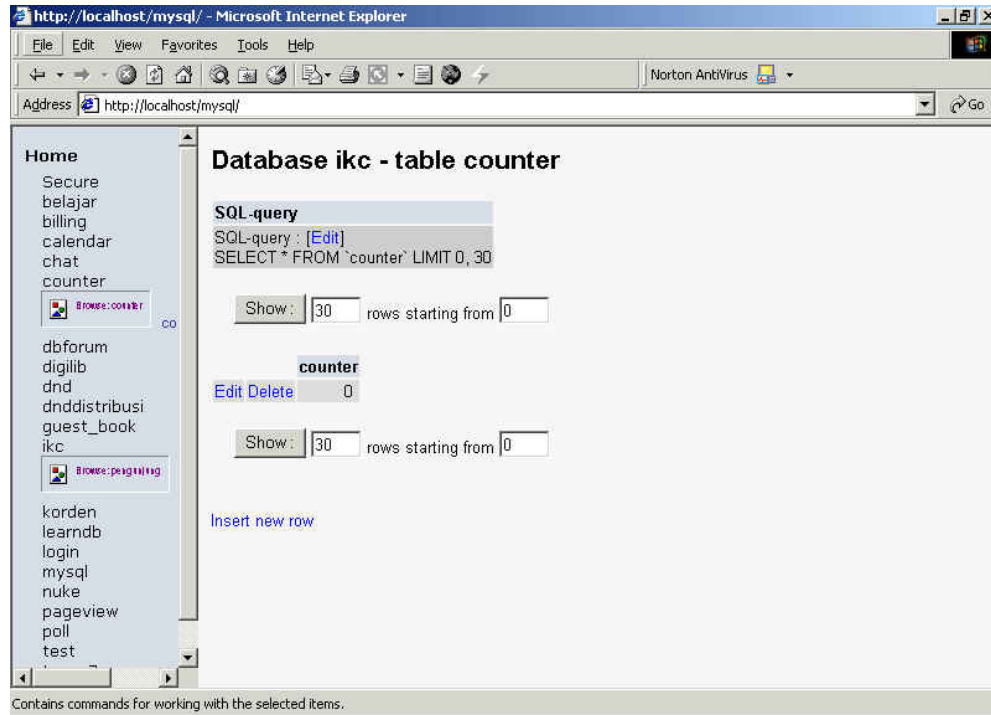
Gambar 20: Counter.php

- **Menggunakan MySql**

Prinsip kerjanya yaitu setiap halaman web di buka php akan memasukkan data ke dalam mysql. Setiap halaman di buka maka nilai yang ada di database mysql di update dinaikkan 1 (+ 1). Untuk lebih jelasnya lihat contoh berikut :

```
CREATE TABLE counter (  
    counter int(5) NOT NULL default '0'  
) TYPE=MyISAM;
```

Buat tabel dengan sintak seperti diatas, atau anda dapat membuatnya dengan phpmyadmin masih dalam database **ikc** , buat tabel bernama **counter** dengan mempunyai satu field bernama **counter**. Isikan angka 0 pada field **counter**.



Gambar 21: Tabel counter

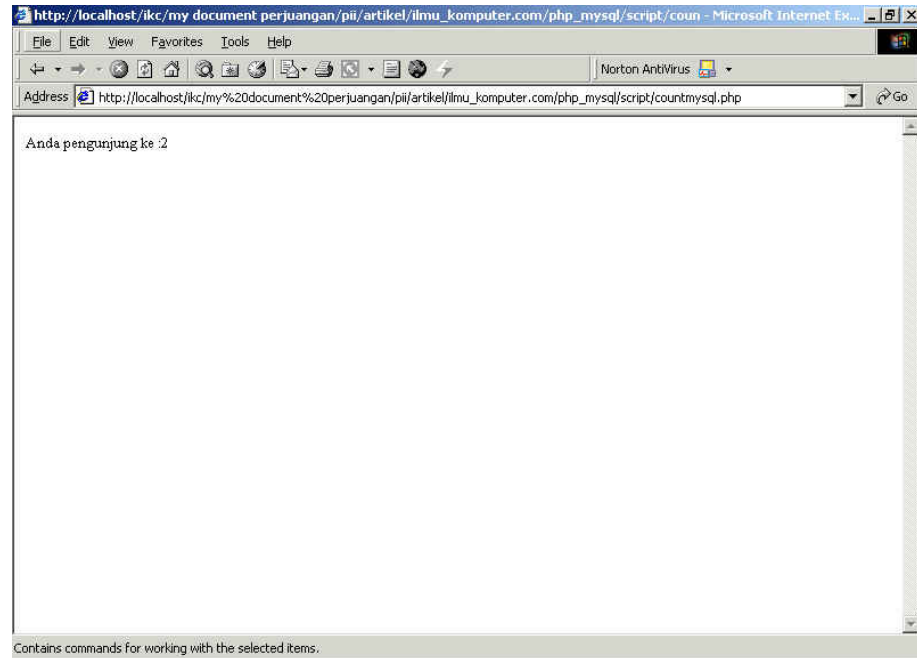
Nilai default 0 (nol) sebagai nilai inisialisasi.

Tabel 14: Source code counter.php

counter.php	
1	<?
2	\$koneksi = mysql_connect("localhost", "root", "");
3	mysql_select_db("ikc", \$koneksi);
4	mysql_query("update counter set counter = counter + 1", \$koneksi);
5	\$query = mysql_query("select counter from counter", \$koneksi);
6	\$row = mysql_fetch_array(\$query);
7	echo "Anda pengunjung ke .:";
8	echo \$row[0];
9	
10	?>

Penjelasan :

- Baris 4 mengupdate data counter yang semula di isi 0 (nol; lihat gambar 19) untuk di tambah 1 nilainya. Jadi nilai counter sekarang ialah 1. Jika ada yang pengunjung lagi maka nilai 1 akan ditambah 1 sehingga menjadi 2 dan seterusnya.
- Baris 4-8 pastinya anda sudah tahu bahwa perintah tersebut untuk menampilkan data.



Gambar 22: Preview counter.php

Nb. Jika source code di atas dijalankan ternyata ada error silahkan mengirim email kepada penulis.

Daftar Pustaka

Wayan FM, Modul Praktikum Pemrograman Internet. D-3 Manajemen Informatika
Fakultas MIPA Brawijaya . 2003. Malang.

Counter.php. webmaster@dony.com.kg

Biografi Penulis



Moh. Safii lahir di Kota Malang , tanggal 12 Oktober 1982. Jenjang SMU di lewatinya di SMU N 1 Malang dan sekarang kuliah di D-3 Manajemen Informatika Semester V Brawijaya Fakultas MIPA. Kebanggaan terbesarnya ialah pernah menjadi anggota Paskibraka 1999 sebagai pengibar bendera di Balaikota Kodya Malang

Mengenal linux sejak tahun 2002. Sebagai penulis artikel di majalah komputer. Artikelnya pernah di terbitkan di berbagai majalah komputer diantaranya PCPlus dan Neotek. Selain aktif menjadi penulis artikel, juga telah menulis buku pertamanya seputar Linux yang di terbitkan oleh Andi Offset.

Informasi lebih lanjut tentang penulis dapat dihubungi lewat email di karebet_asli@telkom.net, sapitenk@telkom.net , m_syafii@telkom.net
<http://sapitenk.cjb.net>

*Sayyidina Ali Karromallahu wajhah berkata :
Jangan melihat siapa yang berbicara tapi lihatlah apa yang di bicarakan.*