

# Relasi Table pada Mysql Menggunakan Keluarga Join

**Mohamad albar**

*mint.argument25@gmail.com*

*http://albar10.wordpress.com*

*Lisensi Dokumen:*

*Copyright © 2003-2007 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.*

Relasi table pada database sangat diperlukan untuk menyederhanakan data-data kita, dan supaya mudah untuk diatur. Ada beberapa cara yang digunakan untuk merelasikan table. Pada pelajaran kali ini saya akan mengajarkan cara relasi database dengan menggunakan keluarga JOIN. Relasi table dengan Join ada 6 macam.

- CROSS JOIN
- STRAIGHT JOIN
- INNER JOIN
- RIGHT JOIN
- LEFT JOIN
- NATURAL JOIN

Sebelum memulai pelajaran kali ini saya asumsikan di computer anda sudah terinstal mysql server, dan phpmyadmin. Jika anda belum punya silahkan download disini untuk phpmyadmin [http://www.phpmyadmin.net/home\\_page/downloads.php](http://www.phpmyadmin.net/home_page/downloads.php) dan untuk mysql download disini <http://www.mysql.com/downloads/mysql/> atau agar lebih mudah download aja yang sudah dipaket, anda bias gunakan wamp download disini <http://www.wampserver.com/en/>.

Untuk mempermudah pelajaran kali ini saya sudah menyiapkan database yang sudah saya export dengan nama produk.sql. silahkan di import melalui phpmyadmin. dan juga saya sudah siapkan versi **dump** nya.

**Note:** Cocokan versi mysql dump anda dengan dump yang saya gunakan kalau tidak database tidak bias di import.

**Note:** Query Mysql bersifat *case-insensitif* sehingga perbedaan huruf kecil dan besar tidak masalah. tetapi untuk nama table, nama database, record, nama\_field tidak berlaku.

Contoh:

Anda membuat table namanya nilai dengan huruf kecil semua. untuk menampilkan isi dari table gunakan query berikut

```
select * from nilai //work
select * from Nilai //don't work
select * from NILAI //don't work
SeLeCt * FroM nilai //work
```

Ok sekarang masuk dulu ke mysql server anda kalau pakai console ketik mysql -u root -p [enter] masukkan password.

Setelah itu masuk ke database produk

```
mysql> use produk
```

Lalu untuk melihat table yang sudah dibuat ketik perintah berikut

```
mysql> show tables;
+-----+
| Tables_in_produk |
+-----+
| jenis             |
| produk            |
+-----+
2 rows in set (0.00 sec)
```

Disitu terdapat dua table yang ingin kita relasikan yaitu table jenis dan produk. Untuk melihat deskripsi dari kedua table tersebut gunakan query berikut

```
mysql> desc produk;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id_produk  | int(11)    | NO   | PRI | NULL    | auto_increment |
| nm_produk  | varchar(40)| YES  |     | NULL    |                 |
| no_jenis   | int(3)     | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
```

```
3 rows in set (0.01 sec)

mysql> desc jenis
-> ;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| no_jenis   | int(11)       | NO   | PRI | NULL    | auto_increment |
| nm_jenis   | varchar(20)   | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
```

Ok sekarang kita perhatikan pada table jenis dan produk terdapat field yang sama yaitu no\_jenis.yang akan kita gunakan untuk merelasikan kedua table berikut.  
 Pada table produk primary key nya adalah id\_produk sedangkan pada table jenis primary key nya adalah no\_jenis.Nah,pada table jenis no\_jenis adalah kunci primer(primary key),sedangkan pada table produk no\_jenis adalah kunci tamu(foreign key).

Sekarang kita lihat isi dari masing-masing table berikut

```
mysql> select * from jenis;
+-----+-----+
| no_jenis | nm_jenis |
+-----+-----+
| 1       | Furniture |
| 2       | Elektronik |
| 3       | FOOD      |
+-----+-----+
```

```
mysql> select * from produk;
+-----+-----+-----+
| id_produk | nm_produk | no_jenis |
+-----+-----+-----+
| 1         | Lemari    | 1        |
| 2         | Kulkas    | 2        |
| 3         | Meja      | 1        |
| 4         | Bangku    | 1        |
| 5         | Televisi  | 2        |
| 6         | Komputer  | 2        |
+-----+-----+-----+
```

Nah pada field/kolom no\_jenis pada table jenis dan produk ,kita bermaksud menghubungkan  
 Komunitas eLearning IlmuKomputer.Com  
 Copyright © 2003-2007 IlmuKomputer.Com

no\_jenis di table jenis dan no\_jenis di table produk, agar angka-angka di field no\_jenis pada table produk diganti dengan huruf pada field nm\_jenis di table jenis. kita bias melakukan hal tersebut dengan merelasi kan kedua table tersebut.

**Query Relasi :** select nama\_table1.nama\_kolom1,nama\_table2.nama\_kolom2  
.... nama\_table(n).nama\_kolom(n) from nama\_table1,nama\_table2..nama\_table(n)

### A. Cara Manual

Baik, cara manual untuk merelasikan kudua table yaitu dengan query seperti berikut

```
mysql> select produk.id_produk, produk.nm_produk, jenis.nm_jenis from
produk, jenis;
```

| id_produk | nm_produk | nm_jenis   |
|-----------|-----------|------------|
| 1         | Lemari    | Furniture  |
| 1         | Lemari    | Elektronik |
| 1         | Lemari    | FOOD       |
| 2         | Kulkas    | Furniture  |
| 2         | Kulkas    | Elektronik |
| 2         | Kulkas    | FOOD       |
| 3         | Meja      | Furniture  |
| 3         | Meja      | Elektronik |
| 3         | Meja      | FOOD       |
| 4         | Bangku    | Furniture  |
| 4         | Bangku    | Elektronik |
| 4         | Bangku    | FOOD       |
| 5         | Televisi  | Furniture  |
| 5         | Televisi  | Elektronik |
| 5         | Televisi  | FOOD       |
| 6         | Komputer  | Furniture  |
| 6         | Komputer  | Elektronik |
| 6         | Komputer  | FOOD       |

18 rows in set (0.00 sec)

Kalau kita lihat output diatas rada aneh ya ☺ karena tidak sesuai yang kita harapkan. terdapat banyak redudansi data pada output diatas nah, untuk mengatasi hal tersebut kita bias gunakan

klausa **where** sehingga query nya jadi seperti berikut

```
mysql> select produk.id_produk,produk.nm_produk,jenis.nm_jenis from
produk,jenis where produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
|      1   | Lemari    | Furniture |
|      2   | Kulkas    | Elektronik |
|      3   | Meja      | Furniture |
|      4   | Bangku    | Furniture |
|      5   | Televisi  | Elektronik |
|      6   | Komputer  | Elektronik |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

Sekarang table sudah muncul sesuai yang diharapkan.ini dikarenakan yang kita relasikan hanya no\_jenis di table produk dan no\_jenis di table jenis.

Ok sekarang sesudah kita paham dengan konsep relasi,kita akan bedakan relasi cara Manual di atas dengan menggunakan Keluarga JOIN

## **B.JOIN TABLE**

### **1.CROSS JOIN**

Pada dasarnya cross join tidak ada bedanya dengan Cara manual di atas yang membedakan adalah tanda **Koma(,)** sesudah kata **from**.query nya seperti berikut.

```
mysql> select produk.id_produk,produk.nm_produk,jenis.nm_jenis from
produk cross join jenis where produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
|      1   | Lemari    | Furniture |
|      2   | Kulkas    | Elektronik |
|      3   | Meja      | Furniture |
|      4   | Bangku    | Furniture |
|      5   | Televisi  | Elektronik |
|      6   | Komputer  | Elektronik |
+-----+-----+-----+
```

Bisa kita lihat tanda koma pada cara manual,di cara Cross join diganti dengan klausa **Cross join**

Oh ia satu lagi tambahan Klausa Cross Join ini selain bias menggunakan klausa **where** juga bisa menggunakan kalusa **On**.

```
mysql> select produk.id_produk,produk.nm_produk,jenis.nm_jenis from
produk cross join jenis on produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
| 1 | Lemari | Furniture |
| 2 | Kulkas | Elektronik |
| 3 | Meja | Furniture |
| 4 | Bangku | Furniture |
| 5 | Televisi | Elektronik |
| 6 | Komputer | Elektronik |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

## 2.STRAIGHT JOIN

Straight Join juga pada dasarnya sama seperti cross join perbedaan yang jelas disini adalah Straight Join **tidak bisa** menggunakan Klausa Kondisi seperti Where atau On

```
mysql> select id_produk,nm_produk,nm_jenis from produk straight join
jenis where produk.no_jenis=jenis.no_jenis;
ERROR 1054 (42S22): Unknown column 'produk.no_jenis' in 'where clause'
mysql> select id_produk,nm_produk,nm_jenis from produk straight join
jenis on produk.no_jenis=jenis.no_jenis;
ERROR 1054 (42S22): Unknown column 'produk.no_jenis' in 'on clause'
```

Bisa kita lihat,jika menggunakan clausa where atau on akan menampilkan pesan error

```
mysql> select id_produk,nm_produk,nm_jenis from produk straight join
jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
| 1 | Lemari | Furniture |
| 1 | Lemari | Elektronik |
| 1 | Lemari | FOOD |
| 2 | Kulkas | Furniture |
| 2 | Kulkas | Elektronik |
| 2 | Kulkas | FOOD |
| 3 | Meja | Furniture |
| 3 | Meja | Elektronik |
| 3 | Meja | FOOD |
```

|   |          |            |
|---|----------|------------|
| 4 | Bangku   | Furniture  |
| 4 | Bangku   | Elektronik |
| 4 | Bangku   | FOOD       |
| 5 | Televisi | Furniture  |
| 5 | Televisi | Elektronik |
| 5 | Televisi | FOOD       |
| 6 | Komputer | Furniture  |
| 6 | Komputer | Elektronik |
| 6 | Komputer | FOOD       |

18 rows in set (0.00 sec)

Jadi Kesimpulannya jangan gunakan Straight join supaya tidak ada penumpukkan/redudansi data

### 3.INNER JOIN

Inner Join konsepnya sama seperti Cross Join Tidak ada perbedaan apapun

```
mysql> select id_produk,nm_produk,nm_jenis from produk inner join jenis
on produk.no_jenis=jenis.no_jenis;
```

| id_produk | nm_produk | nm_jenis   |
|-----------|-----------|------------|
| 1         | Lemari    | Furniture  |
| 2         | Kulkas    | Elektronik |
| 3         | Meja      | Furniture  |
| 4         | Bangku    | Furniture  |
| 5         | Televisi  | Elektronik |
| 6         | Komputer  | Elektronik |

6 rows in set (0.00 sec)

```
mysql> select id_produk,nm_produk,nm_jenis from produk inner join jenis
where produk.no_jenis=jenis.no_jenis;
```

| id_produk | nm_produk | nm_jenis   |
|-----------|-----------|------------|
| 1         | Lemari    | Furniture  |
| 2         | Kulkas    | Elektronik |
| 3         | Meja      | Furniture  |
| 4         | Bangku    | Furniture  |
| 5         | Televisi  | Elektronik |
| 6         | Komputer  | Elektronik |

6 rows in set (0.00 sec)

#### 4.LEFT JOIN

Left Join digunakan untuk menampilkan semua record pada table di sebelah kiri .

```
mysql> select id_produk,nm_produk,nm_jenis from produk left join jenis
on produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
| 1 | Lemari | Furniture |
| 2 | Kulkas | Elektronik |
| 3 | Meja | Furniture |
| 4 | Bangku | Furniture |
| 5 | Televisi | Elektronik |
| 6 | Komputer | Elektronik |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

Kita Lihat, Pada contoh diatas left join akan menampilkan semua record pada table di sebelah kiri yaitu table produk sedangkan table jenis berada disebelah kanan sehingga table jenis akan menyesuaikan .

#### 5.RIGHT JOIN

Kebalikan dari left join ,right join digunakan untuk menampilkan semua record pada table di sebelah kanan.

```
mysql> select id_produk,nm_produk,nm_jenis from produk right join jenis
on produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
| 1 | Lemari | Furniture |
| 3 | Meja | Furniture |
| 4 | Bangku | Furniture |
| 2 | Kulkas | Elektronik |
| 5 | Televisi | Elektronik |
| 6 | Komputer | Elektronik |
| NULL | NULL | FOOD |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

Dapat kita lihat Table jenis akan ditampilkan semua sehingga untuk field/kolom nm\_jenis FOOD karena tidak ada pasangan di table produk maka table produk akan otomatis mengisinya dengan NULL.



**Warning!!**

**Right Join dan Left Join tidak bias menggunakan klausa where sehingga apabila kita gunakan klausa where maka akan muncul pesan error**

## 6.NATURAL JOIN

Natural Join ini mungkin merupakan jawaban bagi query-query yang panjang seperti diatas.karena tanpa menggunakan kalusa where,natural join akan secara otomatis merelasikan table tersebut dengan sangat tepat.

```
mysql> select id_produk,nm_produk,nm_jenis from produk natural join
jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
|          1 | Lemari    | Furniture |
|          2 | Kulkas    | Elektronik |
|          3 | Meja      | Furniture |
|          4 | Bangku    | Furniture |
|          5 | Televisi  | Elektronik |
|          6 | Komputer  | Elektronik |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

Jika menggunakan where

```
mysql> select id_produk,nm_produk,nm_jenis from produk natural join
jenis where produk.no_jenis=jenis.no_jenis;
+-----+-----+-----+
| id_produk | nm_produk | nm_jenis |
+-----+-----+-----+
|          1 | Lemari    | Furniture |
|          2 | Kulkas    | Elektronik |
|          3 | Meja      | Furniture |
|          4 | Bangku    | Furniture |
|          5 | Televisi  | Elektronik |
|          6 | Komputer  | Elektronik |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

**Note:***Natural join tidak bias menggunakan klausa on ,sehingga jika kita meggunakan klausa tersebut akan muncul pesan error.*

## BIOGRAFI PENULIS



**Mohamad Albar**, Lahir di Jakarta, 25 Mei 1992. Menamatkan SMU di SMU Negeri 113 Jakarta, dan sekarang sedang menjalankan study S1 di Universitas Gunadarma.

Informasi Lebih Lanjut tentang penulis dapat melalui.

URL : <http://heresy7.com>

BLOG : <http://albar10.wordpress.com>

EMAIL : [mint.argument25@yahoo.com](mailto:mint.argument25@yahoo.com)