

SELEKSI DATA PADA MYSQL

Muhamad Zaelani

Zaelani@raharja.info

Abstrak

Pengolahan manajemen database tidak luput dengan perintah select, dimana perintah select ini merupakan perintah utama dalam memanajemen database tersebut, mungkin kita hanya tau perintah select itu sekedar perintah untuk mendapatkan nilai yang kita inginkan saja, memang betul perintah select merupakan perintah untuk melakukan pengambilan data dari satu tabel atau lebih. Perintah select mempunyai banyak macamnya seperti perintah select filter, limit, pengelompokan data, penggabungan data dan sebagainya.

Kata Kunci: MySQL, Select, Database.

Pendahuluan

Sejauh ini kita telah mempelajari mengenai tentang pembuatan database, pengenalan database, serta bagaimana memanipulasi data tersebut. tahap selanjutnya yaitu bagaimana caranya mengambil data yang kita inginkan dari satu tabel ataupun lebih. Perintah yang digunakan merupakan perintah *select*. Perintah ini banyak dijumpai di setiap pembuatan program, maupun manajemen databases.

Banyak yang termasuk kedalam perintah select, seperti perintah alias, limit, kemudian group by dan sebagainya. Untuk membatasi pengetahuan perintah select, pada artikel ini yang akan dibahas merupakan bagaimana mengambil data dari satu tabel atau lebih dengan ditambahkan perintah select, limit, maupun group by. Untuk lebih jelasnya kita langsung saja ke dalam pembahasan yang akan di tuju.

Lisensi Dokumen:

Copyright © 2008-2017 ilmuti.org

Seluruh dokumen di ilmuti.org dapat digunakan, dimodifikasi dan disebarikan 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 ilmuti.org

Pembahasan

1. Pengertian Perintah *Select*

Perintah *Select* merupakan perintah untuk mengambil dan menampilkan data dari *databases*, perintah yang digunakan yaitu perintah *select*. Mungkin kalian juga telah mengetahui apa itu perintah *select* sebelumnya. Perintah *select* mempunyai beberapa klausa yang bersifat opsional, ini berarti bahwa kita dapat menggunakan bagaian tersebut atau tidak, namun biasanya pada saat tertentu kita membutuhkannya. Klausa tersebut seperti *LIMIT*, tidak dapat ditempatkan sebelum klausa *WHERE*.

Studi Kasus :

Dibawah ini merupakan perintah *select* yang dimana penulis memilih database dokter yang akan di tampilkan. Adapun perintah nya sebagai berikut :

```
Select * From dokter;
```

```
mysql> use rumah_sakit;
Database changed
mysql> select *from dokter;
```

id_dokter	nama_dokter	spesialis	tanggal_lahir	alamat	no_telp
D001	Muhammad zaelani	Penyakit Dalam	1996-05-26	rajeg tangerang	89998855
D002	Rizki atmaharun	Umum	1990-02-21	Cikokol Tangerang	554488
d003	Mala Fazriah	Gigi	1989-07-02	kp gembong, sukamaju	877998872
D004	Abdullah nur Alamsyah	THT	1980-11-24	Bogor	875564214

4 rows in set (0.07 sec)

Perintah diatas berguna untuk menampilkan semua data yang berada di tabel dokter. Adapun perintah asterisk (*) merupakan perintah untuk menampilkan semua kolom dalam tabel yang bersangkutan.

Jika suatu saat kita hanya memerlukan nama dokter beserta spesialisnya saja maka perintah yang digunakan yaitu seperti dibawah ini :

```
Select nama_dokter, spesialis from dokter ;
```

```
mysql> select nama_dokter, spesialis from dokter;
+-----+-----+
| nama_dokter | spesialis |
+-----+-----+
| Muhammad zaelani | Penyakit Dalam |
| Rizki atmaharun | Umum |
| Mala Fazriah | Gigi |
| Abdullah nur Alamsyah | THT |
+-----+-----+
4 rows in set (0.00 sec)
```

Creative By Muhammad Zaelani

Pada perintah select diatas hanya menampilkan dua kolom yaitu nama dokter dan spesialisnya saja.

a. Menggunakan Alias

Perintah alias merupakan perintah untuk memberi nama pada tabel pada saat menentukan kepemilikan pada kolom dalam perintah *select*. Ada dua perintah alias yang dapat digunakan seperti perintah alias yang digunakan untuk memberi informasi tabel atau database dan mengganti nama pada tabel .

Perhatikan contoh gambar di bawah ini :

```
mysql> select
-> pasien.id_pasien, pasien.nama_pasien, pasien.umur
-> from
-> pasien;
+-----+-----+-----+
| id_pasien | nama_pasien | umur |
+-----+-----+-----+
| p002 | afikah setia dewi | 17 |
| p003 | Putri wulan | 15 |
| p005 | putra dwi simorangkir | 9 |
| p011 | Darma putra atmadja | 17 |
| p033 | Cinta ari kusumawati | 18 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

Creative By Muhammad Zaelani

Perintah diatas dapat disederhanakan dengan menggunakan perintah select dibawah ini :

Lisensi Dokumen:

Copyright © 2008-2017 ilmuti.org

Seluruh dokumen di ilmuti.org dapat digunakan, dimodifikasi dan disebarakan 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 ilmuti.org

```
mysql> select a.id_pasien, a.nama_pasien, a.umur
-> from
-> pasien a;
```

id_pasien	nama_pasien	umur
p002	afikah setia dewi	17
p003	Putri wulan	15
p005	putra dwi simorangkir	9
p011	Darma putra atmadja	17
p033	Cinta ari kusumawati	18

5 rows in set (0.00 sec)

creative by
Muhammad Zaetani

Dapat disimpulkan bahwa (a) merupakan alias (nama lain) dari tabel pasien.

Selain untuk menjadikan alias tersebut menjadi tabel atau database, alias dapat digunakan untuk mengganti nama pada tabel, seperti gambar dibawah ini :

```
mysql> select
-> id_pasien as 'ID Pasien', nama_pasien as 'Pasien', tanggal_lahir as 'Tanggal Lahir'
-> from
-> pasien;
```

ID Pasien	Pasien	Tanggal Lahir
p002	afikah setia dewi	2000-07-25
p003	Putri wulan	2004-12-28
p005	putra dwi simorangkir	2005-03-07
p011	Darma putra atmadja	2000-02-02
p033	Cinta ari kusumawati	1999-09-09

5 rows in set (0.00 sec)

creative by
Muhammad Zaetani

Dapat dilihat bahwa pada judul kolom yang seharusnya merupakan nama kolom tersebut menjadi nama yang kita inginkan seperti Id Pasien yang nama dari kolom tersebut adalah id_pasien dan kolom lainnya.

b. Menyaring data (*Filter*)

Data selain ditampilkan harus menyaring data tersebut sesuai dengan kriteria yang kita inginkan, perintah yang digunakan merupakan perintah *filter*. Sebelum menggunakan *filter* kita harus mendefinisikan terlebih dahulu kondisi seperti apa yang akan dijadikan kriteria untuk proses penyaringan.

Studi Kasus :

Kita akan menampilkan data dalam tabel pasien yang dimana umur pasien di atas sama dengan 17. Maka perintah nya sebagai berikut :

```
mysql> select
-> id_pasien, nama_pasien, tanggal_lahir, umur, alamat
-> from pasien
-> where umur >= 17;
```

id_pasien	nama_pasien	tanggal_lahir	umur	alamat
p002	afikah setia dewi	2000-07-25	17	taman nuansa indah
p011	Darma putra atmadja	2000-02-02	17	cengkareng jakarta barat
p033	Cinta ari kusumawati	1999-09-09	18	Kebon jeruk jakarta

3 rows in set (0.00 sec)

creative By
Muhammad Zaenal

Pada perintah diatas akan mendefinisikan kondisi ($umur \geq 17$) dalam klausa WHERE . maka yang akan ditampilkan merupakan data yang memenuhi syarat kriteria yang kita inginkan saja.

c. Membatasi Data (*Limit*)

Perintah LIMIT didukung oleh MySQL yang meminta server MYSQL untuk mngembalikan sejumlah data yang diinginkan saja, dengan kata lain jumlah baris data yang ditampilkan dapat kita batasi sesuai yang kita inginkan.

Dalam penggunaannya, LIMIT dapat memiliki satu atau dua buah argumen seperti dibawah ini

LIMIT <jumlah data>

LIMIT <jumlah data yangng diabaikan>, <jumlah data>

<jumlah data> merupakan banyaknya data yang ingin ditampilkan, sedangkan <jumlah data yang diabaikan> meruppakan banyaknya baris data dalam table yang akan dilewati atau diabaikan.

Untuk memahami perintah limit, penulis akan memberikan studi kasusnya sebagai berikut :

Lisensi Dokumen:

Copyright © 2008-2017 ilmuti.org

Seluruh dokumen di ilmuti.org dapat digunakan, dimodifikasi dan disebarcan 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 ilmuti.org


```
mysql> select * from obat;
+-----+-----+-----+-----+
| id_obat | nama_obat | jml_obat | harga |
+-----+-----+-----+-----+
| B001    | Bodrexin  | 3         | 15000 |
| B011    | Kalbine   | 12        | 10000 |
| B023    | Captrovil | 3         | 13000 |
| B025    | gilbanclamide | 10      | 10000 |
| B027    | Paracetamol | 5        | 12000 |
+-----+-----+-----+-----+
5 rows in set (0.05 sec)

mysql> select * from obat limit 4;
+-----+-----+-----+-----+
| id_obat | nama_obat | jml_obat | harga |
+-----+-----+-----+-----+
| B001    | Bodrexin  | 3         | 15000 |
| B011    | Kalbine   | 12        | 10000 |
| B023    | Captrovil | 3         | 13000 |
| B025    | gilbanclamide | 10      | 10000 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from obat limit 2, 5;
+-----+-----+-----+-----+
| id_obat | nama_obat | jml_obat | harga |
+-----+-----+-----+-----+
| B023    | Captrovil | 3         | 13000 |
| B025    | gilbanclamide | 10      | 10000 |
| B027    | Paracetamol | 5        | 12000 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

bisa dilihat dari gambar diatas, perintah yang pertama merupakan perintah untuk menampilkan semua data yang ada di tabel obat, kemudia perintah yang kedua yaitu perintah untuk menampilkan data obat dengan hanya menampilkan data 4 saja dimulai dari data yang pertama. Sedangkan perintah yang ke 3 merupakan perintah untuk menampilkan dari data ke 2 sampai dengan data yang ke 5.

Lisensi Dokumen:

Copyright © 2008-2017 ilmuti.org

Seluruh dokumen di ilmuti.org dapat digunakan, dimodifikasi dan disebarakan 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 ilmuti.org

d. Menggunakan Distinct

Menggunakan perintah Distinct merupakan untuk menghindari duplikasi data yang kita buat. Jika suatu query menghasilkan sejumlah baris data yang mengandung baris-baris data yang sama, maka kita dapat menghilangkan duplikat dari baris tersebut dengan menyertakan perintah *Select Distinct* <...>

Sebagai contoh sebagai berikut :

```
mysql> select *from rekam_medis;
```

id_RM	id_pasien	tanggal_periksa	id_dokter	diagnosa
RM01	p011	2017-04-01	D001	pusing, lemas, batuk batuk
RM02	P033	2017-04-10	D002	Mual, demam,
RM03	p002	2017-03-12	d001	jantung lemas, denyut melemah
RM04	p033	2017-03-22	d001	Paru paru basah
RM05	p011	2017-03-20	d003	Gigi berlubang
RM06	p005	2017-04-20	d003	Gigi bengkok
RM07	p002	2017-04-05	d004	telinga mendengung

```
7 rows in set (0.00 sec)
```

```
mysql> select id_dokter from rekam_medis;
```

id_dokter
D001
D002
d001
d001
d003
d003
d004

```
7 rows in set (0.00 sec)
```

Creative By
Muhammad Zakiandi

Dapat dilihat bahwa hasil dari id_dokter terdapat data yang rangkap atau data yang sama. Hal ini sering dinamakan dengan duplikasi data. Untuk mengatasinya yaitu dengan menggunakan perintah DISTINCT, seperti dibawah ini :

```
mysql> select distinct id_dokter from rekam_medis;
```

id_dokter
D001
D002
d003
d004

```
4 rows in set (0.00 sec)
```

Kesimpulannya yaitu perintah ini memastikan bahwa setiap baris data yang ditampilkan bersifat unik atau berbeda dengan yang lainnya.

Penutup

Dapat dipahami perintah select merupakan perintah yang sangat penting untuk mendapatkan informasi dari database yang kita miliki, banyak klausa yang digunakan dalam perintah select ini, seperti klausa where, limit dan sebagainya. Penyeleksian data bukan hanya sekedar mendapatkan data nya saja, tetapi banyak kriteria yang harus diambil untuk memenuhi data yang dibutuhkan. Semoga dengan terbitnya artikel ini dapat membantu mengetahui bagaimana penggunaan perintah select tersebut. dan semoga artikel ini dapat memberikan motivasi kepada pembelajar agar dapat terus meningkatkan pengetahuannya dalam dunia database.

Referensi

- [1] Referensi Pribadi
- [2] Raharjo, Budi 2015 “Belajar otodidak MySQL. Bandung : Informatika

Biografi



Muhammad Zaelani, Lahir di Tangerang, 29 Mei 1996. Anak terakhir dari 4 bersaudara, hobby bermain komputer, baca buku, buku yang sekarang dipelajari mengenai Administrasi Database, DBMS Programing. Kuliah di Perguruan Tinggi daerah Tangerang, mengambil jurusan Sistem Informasi dengan kosentrasi Sistem Informasi Manajemen tahun angkatan 2014, fokus di dalam kuliah kepada sistem database. Aktif dalam berorganisasi. Kompetensi yang dimiliki DBMS Programing, Sql Server, dasar PHP, jaringan komputer, troubleshooting komputer/laptop. Bekerja di Unit Transfusi Darah PMI Kabupaten Tangerang bagian administrasi pelayanan dari 2015 hingga sekarang. Email : Zaelani@raharja.info

Lisensi Dokumen:

Copyright © 2008-2017 ilmuti.org

Seluruh dokumen di ilmuti.org dapat digunakan, dimodifikasi dan disebarikan 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 ilmuti.org