**PERINTAH DASAR SQL**

[October 5, 2014](https://sidkanhakim.wordpress.com/2014/10/05/perintah-dasar-sql/) [irfanhakim96](https://sidkanhakim.wordpress.com/author/irfanhakim96/) [PRAKTIKUM DBD](https://sidkanhakim.wordpress.com/category/praktikum-dbd/)

Assalamualaikum…………..,

Di Postingan kali ini ane pingin berbegi ilmu dengan agan-agan semua tentang perintah dasar SQL, sebenarnya banyak sekali perintah-perintah yang ada di dalam SQL. tapi untuk postingan kali ini ane batasi seputar membuat hubungan (relasi) antar tabel atau dalam bahasa kerennya reference, memasukan data (insert), menampilkan data, manghapus data, memodifikasi data, dan pengurutan data. ok gan langsung aja kita mulai:

* MEMBUAT HUBUNGAN ANTAR TABEL (REFERENCES)

untuk membuat hubungan antar tabel, ada beberapa syarat yang harus dipenuhi, yang pasti tentunya kita harus memiliki dua atau lebih tabel yang akan dihubungkan, berikutnya antara tabel yang ingin dihubungkan  kita harus memiliki sebuah *field*atau kolom yang mempunyai tipe data yang sama. bingung ya gan maksudnya ap????? hehehehehe………, sama gan ane awalnya juga merasakan hal yang sama, ok kurang lebih maksudnya adalah ada satu field yang berfungsi sebagai primery key yang dihubungkan dengan tabel lain sehingga di tabel lain field tersebut menjadi foreign key (key tamu) yang kemudian field tersebut ibaratnya seperti jembatan yang menghubungkan antar dua tabel…….., untuk lebih jelasnya nanti di bagian dua akan ada contohnya…..,

ini ane dapat sumber dari<http://dwirandyherdinanto.blogspot.com/2014/01/membuat-relasi-tabel-di-mysql.html>

Apa itu Relasi Tabel ?  
Relasi tabel adalah hubungan antara beberapa tabel. Relasi antar tabel dihubungkan oleh**primary key** dan **foreign key**  
Apa itu primary key dan foreign key ?  
**primary key** adalah suatu atribut yang tidak hanya mengindentifikasi secara unik suatu kejadian tetapi juga mewakili setiap kejadian dari suatu entitas  
**foregin key** adalah atribut yang melengkapi relationship dan menunjukan hubungan antara tabel induk dengan tabel anak. foreign key ditempatkan pada tabel anak

untuk membuat relationship maka masing-masing tabel harus memiliki primary key dan foreign key untuk dapat menghubungkan antara tabel induk dengan tabel anak. Sehingga diperlukan teknik **normalisasi** terlebih dahulu sebelum membuat relationship antar tabel.  
**Normalisasi**merupakan proses pengelompokan data elemen menjadi tabel yang menunjukan entitas dan relasinya

* MEMASUKKAN DATA

memasukan data ke dalam sebuah tabel dapat dilakukan dengan mengguankan perintah INSERT. ada beberapa cara penulisan perintah yang dapat digunakan untuk memasukan data. diantaranya adalah

conto tabel : tabel data

| no\_id | nama | alamat |

versi 1 :

**insert into data values (1, ‘harir’, ‘tulungagun’);**

versi 2

**insert** **into data (no\_id , nama , alamat ) values (1 , ‘harir’, ‘tulungagung);**

* MENAMPILKAN DATA

dalam bahasa SQL perintah yang digunakan untuk menampilkan data adalah**SELECT.**berikut adalah tata cara penulisannya:

untuk menampilkan satu atau lebi tabel :

**select nama\_kolom from nama\_tabel;**

// untuk menampilkan dua atau lebih kolom maka tambahkan tanda koma (,) setelah nama kolom dan tulis nama kolom yang akan ditampilkan. contoh :

**select kolom1, kolom2, kolom3, from nama\_tabel;**

untuk menampilkal semua tabel :

**select \* from nama\_tabel;**

* MENGHAPUS DATA :

perintah SQL untuk menghapus data adalah DELETE. tata cara penulisannya adalah :

untuk menghapus semua baris yang ada dalam tabel

**delete from nama\_tabel;**

untuk menghapus satu atau beberapa baris yang kita inginkan :

**delete from nama\_tabel where kolom = katakunci;**

* MODIFIKASI DATA

pada postingan ini akan dibahas tentang perintah **UPDATE,**seperti pengertian asalnya update artinya memperbarui, jadi perintah update digunakan untuk memperbarui data yang sudah ada. tata cara penulisannya sebagai berikut :

**update namatabel set namakolom = isidata where namakolom = katakunci;**

* PENGURUTAN DATA

pengurutan data dalam SQL mengguankan perintah **ORDER BY**yang diletakkan setelah perintah menampilkan data. ada dua macam pengurutan yaitu ascending (dari paling kecil ke paling besar) dan descending (dari paling besar ke paling kecil). berikut adalah tata cara penulisan perintah pengurutan data :

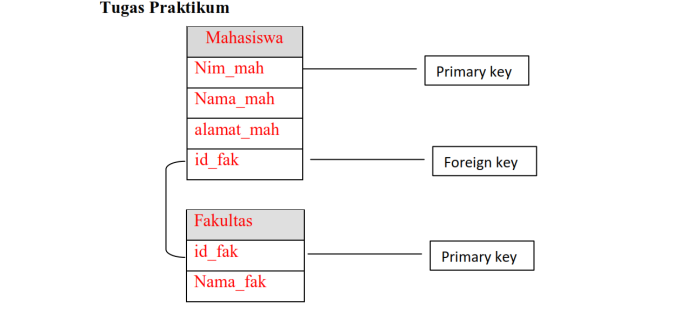
**select \* from nama\_tabel order by nama\_kolom;**

pada perintah di atas data akan diurutkan secara ascending. jika kita ingin mengurutkan data secara descending maka tambahkan perintah **desc**  sebelum tanda titik koma (;).

**TUGAS PRAKTIKUM**

pada penjelasan di bawah ini ane mengguankan mysql untuk mengeksekusi bahasa sql………….,

buatlah tabel seperti gambar di bawah ini :

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_18_19_14_220.png)

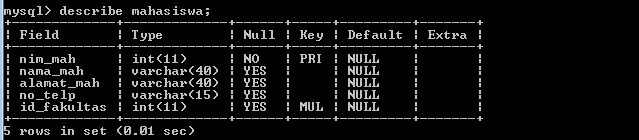
* Membuat Tabel

langkah pertama yang harus kita lakukan adalah membuat kedua tabel di atas.

source code untuk membuat tabel mahasiswa :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | create table mahasiswa (nim\_mah integer primary key,    nama\_mah varchar(40),    alamat\_mah varchar (40),     no\_telp varchar (15),    id\_fakultas integer(11)); |

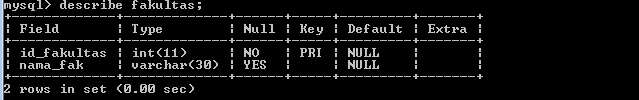
sehingga hasilnya akan seperti ini

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_02_50_309.png)

kemudian buatlah tabel fakultas dengan source code seperti di bawah ini

create table fakultas (id\_fakultas integer(11) primary key, nama\_fak varchar(30));

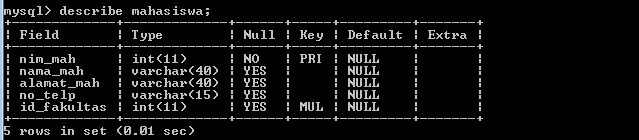
sehingga hasilnya akan seperti gambar dibawah ini

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_10_33_509.png)

setelah itu buatlah relasi antara kedua tabel tersebut dengan cara menulis sourcecode seperti dibawah ini:

|  |  |
| --- | --- |
| 1  2  3 | alter table mahasiswa add foreign key (id\_fakultas)    references fakultas(id\_fakultas) on update cascade on delete cascade; |

sehingga setelah menjalankan perintah di atas tabel mahasiswa akan menjadi seperti di bawah ini :

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_02_50_309.png)

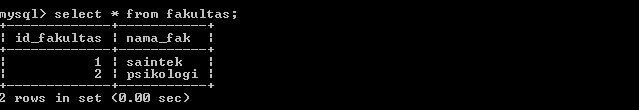
// tanda **MUL**pada id\_fakultas merupakan tanda bahwa dua tabel tersebut sudah direlasikan.

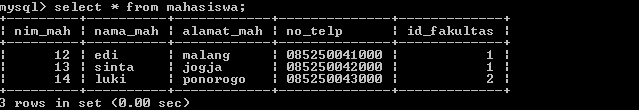
* MEMASUKAN DATA

tulislah source code di bawah ini :

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | insert into fakultas values (1, 'saintek');    insert into fakultas values (2, 'psikologi');    insert into mahasiswa values (12, 'edi', 'malang', '085250041000', 1);    insert into mahasiswa values (13, 'sinta', 'jogja', '085250042000', 1);    insert into mahasiswa values (14, 'luki', 'ponorogo', '085250043000', 2); |

sehingga tabel akan terisi seperti gambar di bawah ini :

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_21_11_01_632.png)

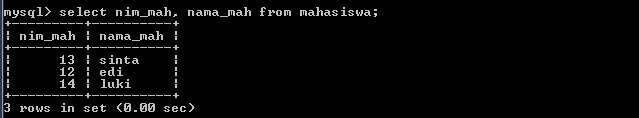
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_21_03_38_322.png)

* MENAMPILKAN DATA BERUPA NIM DAN NAMA

untuk menampilkan data berupa NIM dan nama kita mengguanakan sourcecode berikut:

|  |  |
| --- | --- |
| 1 | select nim\_mah , nama\_mah from mahasiswa; |

sehingga hasilnya akan seperti gambar di bawah ini.

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_35_12_850.png)

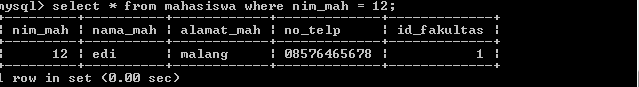
* MENAMPILKAN DATA BERDASARKAN NIM SALAH SATU MAHASISWA

ketikan sourcecode berikut:

ambil contoh data yang akan ditampilkan adalah data dengan nim = 12;

|  |  |
| --- | --- |
| 1 | select \* from mahsiswa where nim\_mah = 12; |

sehingga tampilan yang dihasilkan adalah :

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_37_17_763.png)

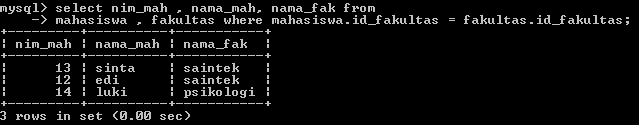
* MENAMPILKAN DATA DARI DUA TABEL

misalnya untuk menampilkan nim, nama dari tabel mahasiswa dan nama fakultas  dari tabel fakultas.

berikut adalah sourcecodenya…,:

|  |  |
| --- | --- |
| 1  2  3 | select nim\_mah, nama\_mah, fakultas from mahasiswa , fakultas    where mahasiswa.id\_fakultas = fakultas.idfakultas; |

maka hasilnya akan seperti ini :

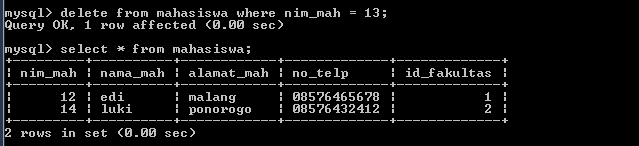
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_39_50_489.png)

* MENGHAPUS DATA

di sini kita mengambil contoh data yang akan dihapus adalah data denngan nim = 13.

berikut sourcecodenya

|  |  |
| --- | --- |
| 1 | delete from mahasiswa where nim\_mah = 13; |

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_41_02_623.png)

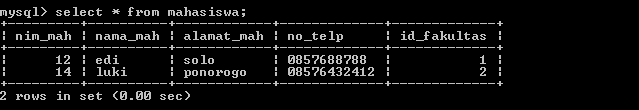
* UPDATE DATA

data yang akan diubah adalah data dengan nim 12, kita ingin memperbarui alamat menjadi solo dan no\_telp menjadi 0857688788. lakukan pembaruan satu per satu……,

berikut sourcecodenya :

|  |  |
| --- | --- |
| 1  2  3 | update mahasiswa set alamat\_mah = 'solo' where nim\_mah = 12;    update mahasiwa set no\_telp = '0857688788' where nim\_mah = 12; |

sehingga akan tampil data sebagai berikut :

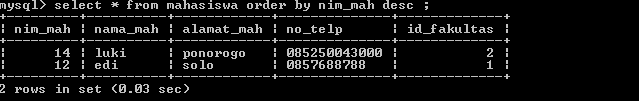
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_20_47_56_643.png)

* MENGURUTKAN DATA

menampilkan data mahasiswa dari besar ke kecil berdasarkan nim mahasiswa;

tuliskan kode burikut:

|  |  |
| --- | --- |
| 1 | select \* from mahasiswa order by nim\_mah desc; |

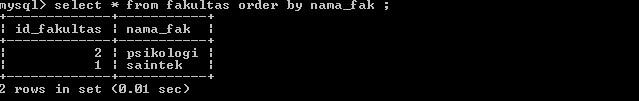
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_22_16_23_210.png)

menampilkan data dari fakultas secara ascending berdasarkan kolom kedua.

sourcecodenya :

|  |  |
| --- | --- |
| 1 | select \* from fakultas order by nama\_fak |

sehingga hasilnya sebagai berikut :

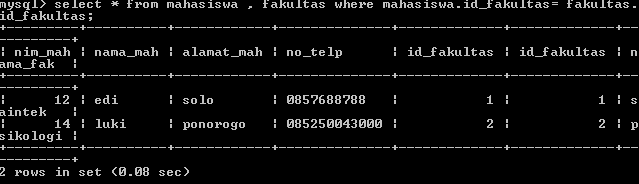
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_22_20_27_81.png)

* MENAMPILKAN SELURUH DATA DARI MAHSISWA DAN FAKULTAS

sourcecodenya :

|  |  |
| --- | --- |
| 1 | select \* from mahasiswa, fakultas where mahasiswa.id\_fakultas = fakultas.id\_fakultas. |

sehingga hasilnya akan seperti gambar di bawah ini :

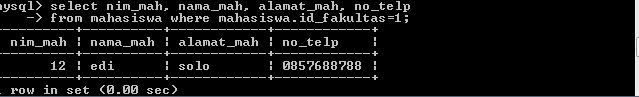
[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_22_26_59_595.png)

* MENAMPILKAN DATA BERDASARKAN SYARAT TERTENTU

menampilkan data mahasiswa yang berada di fakultas saintek.,  berikut sourcecodenya :

|  |  |
| --- | --- |
| 1 | select nim\_mah , nama\_mah , alamat\_mah from mahasiswa where id\_fakultas = 1; |

karena yang terhubung dengan tabel fakultas adalah id\_fakultas maka kita menggunakan id\_fakultas sebagai penghubung.

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_23_18_15_898.png)

* MENGHAPUS DATA

menghapus fakultas saintek dari tabel, sourcecodenya:

|  |  |
| --- | --- |
| 1 | delete from fakultas where nama\_fak = saintek; |

dan hasilnya :

[](https://sidkanhakim.files.wordpress.com/2014/10/capture_2014_10_05_22_46_58_28.png)

ada yang aneh???????

ya, ada biasanya pada saat penghapusan suatu kolom tertentu yang sudah berelasi maka akan muncul ERROR, hal itulah yang saya dapatkan ketika praktikum di lab…., sekedar perkiraan saya mungkin penyebabnya adalah karena tambahan on delete cascade dan on update cascade yang ada pada saat pembuatan relasi….., berikut penjelasan lengkapnya :

1. RESTRICT, Jika tabel anak berisi nilai dalam kolom yang mengkait yang nilainya sama dengan di kolom terkait pada tabel induk, baris dalam tabel induk tidak bisa dihapus, dan nilai di kolom terkait tidak dapat diupdate.  
ini adalah opsi default jika klausa on delete atau on update tidak dispesifikasikan.

2. CASCADE, Baris-baris dalam tabel anak yang berisi nilai-nilai yang juga terdapat dalam kolom terkait dari tabel induk dihapus ketika barisbaris yang berkaitan dihapus dari tabel induk. Baris-baris dalam tabel anak yang berisi nilai-nilai yang juga terdapat dalam kolom terkait dari tabel induk diupdate ketika nilai-nilai yang berkaitan diupdate dalam tabel induk.

3. SET NULL, Nilai-nilai dalam kolom yang mengkait dari tabel anak diset ke NULL saat baris-baris dengan data terkait dalam tabel induk dihapus dari tabel induk atau ketika data terkait dalam tabel induk diupdate. Untuk menggunakan opsi ini, semua kolom-kolom yang mengkait dalam tabel anak harus mengijinkan nilai NULL.

4. NO ACTION Tidak ada aksi yang diambil dalam tabel anak ketika baris-baris dihapus dari tabel induk atau nilai-nilai dalam kolom terkait dalam tabel induk diupdate.

5. SET DEFAULT Nilai-nilai dalam kolom-kolom yang mengkait dari tabel anak diset ke nilai default mereka ketika baris-baris dihapus dari tabel induk atau kolom terkait dari tabel induk diupdate.

sumber : <http://dwirandyherdinanto.blogspot.com/2014/01/membuat-relasi-tabel-di-mysql.html>

**PERBANDINGAN QUERY POSTGRESQL DAN MYSQL**

* MEMBUAT RELASI :

cara penulisan pembuatan hubungan antar tabel antara PSQL dan MYSQL sama, hanya saja pada MYSQL banyak variasi yang bisa digunakan.

* PERINTAH INSERT

secara keseluruhan sama cara penulisan insert antara PSQL  dan MYSQL

* MENGHAPUS DATA

cara menghapus data jug sama.

* UPDATE DATA

secara keseluruhan sama…..

* SORTING

sama-sama menggunakan order by…..

* pada hubungan antar tabel mungkin PSQl lebih ketat dalam merelasikan dua tabel sedangka MYSQL banyak pilihan yang dapat digunakan untuk opsi-opsi relasi.

**KESIMPULAN**

sejauh pemahaman saya bahwa mysql dan postgresql secara keseluruhan sama dalam hal query untuk merelasikan tabel, menginsert data, mengupdate data, menyortir data…..,  sehinggan secara keseluruhan dalam laporan kali ini tidak banyak perbedaan yang saya temukan……,

**SARAN**

mungkin saran saya pada saat praktikum lebih diperbanyak lagi materi yang diajarkan………….,