

Modul 8

Relasi Tabel

Tujuan

1. Membuat relasi tabel
2. Membuktikan integritas data dengan tabel yang ter-relasi

Dasar Teori

Relasional Database dan RDBMS, Relational Database sebenarnya adalah salah satu konsep penyimpanan data, sebelum konsep database relasional muncul sebenarnya sudah ada dua model database yaitu Network Database dan Hierarchie Database. Dalam database relasional, data disimpan dalam bentuk relasi atau tabel dua dimensi, dan antar tabel satu dengan tabel lainnya terdapat hubungan atau relationship, sehingga sering kita baca di berbagai literatur, database didefinisikan sebagai “kumpulan dari sejumlah tabel yang saling hubungan atau keterkaitan”. Nah, kumpulan dari data yang diorganisasikan sebagai tabel tadi disimpan dalam bentuk data elektronik di dalam hardisk komputer. Untuk membuat struktur tabel, mengisi data ke tabel, mengubah data jika diperlukan dan menghapus data dari tabel diperlukan software.

Software yang digunakan membuat tabel, isi data, ubah data dan hapus data disebut Relational Database Management System atau dikenal dengan singkatan RDBMS sedangkan perintah yang digunakan untuk membuat tabel, isi, ubah dan hapus data disebut perintah SQL yang merupakan singkatan dari Structure Query Language. Jadi, setiap software RDBMS pasti bisa digunakan untuk menjalankan perintah SQL.

Sebenarnya fungsi RDBMS bukan cuma buat tabel, isi data, ubah dan hapus data, untuk manajemen data dalam skala besar dan agar bisa mendukung proses bisnis yang kontinyu dan real time. Suatu RDBMS dituntut untuk mempunyai kemampuan manajemen user dan keamanan data, backup dan recovery data serta kemampuan lainnya yang berkaitan dengan kecepatan pemrosesan data (performance).

Primary key

Constraint ini membentuk key yang unik pada suatu table sehingga isi tiap recordnya dengan mudan dibedakan.

Contoh:

```
mysql> alter table student add constraint id primary key(nim);
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc student;
```

Field	Type	Null	Key	Default	Extra
nim	char(11)	NO	PRI		
nama	varchar(15)	NO		NULL	
alamat	varchar(25)	YES		NULL	
telepon	varchar(20)	YES		NULL	
email	varchar(25)	YES		NULL	

Pada contoh diatas (table student), field nim bertindak sebagai constraint primary key, kenapa nim yang digunakan sebagai primary key? Karena nim merupakan satu-satunya field pembeda yang ada pada table tersebut, nama setiap murid mungkin saja ada yang sama, namun nim setiap murid sudah pasti berbeda-beda, maka dari itu nim digunakan sebagai primary key pada table student ini.

Foreign key

Constraint ini berada pada suatu table namun pada kolomnya tersebut juga dimiliki oleh teble lain sebagai primary key.

Contoh:

Misalkan kita disuruh membuat tabel dosen_pembimbing dengan tipe innodb dan relasikan ke tabel mahasiswa mengambil key nim dari tabel mahasiswa sebagai foreign key di tabel dosen_pembimbing.

Pertama kita harus membuat table mahasiswa dengan tipe innodb dan nim sebagai primary keynya sebagai berikut ini.

```
mysql> use akademik;
Database changed
mysql> create table mahasiswa (
  -> nim varchar(20) not null,
  -> nama varchar(25),
  -> primary key(nim));
Query OK, 0 rows affected (0.08 sec)

mysql> desc mahasiswa;
```

Field	Type	Null	Key	Default	Extra
nim	varchar(20)	NO	PRI	NULL	
nama	varchar(25)	YES		NULL	

2 rows in set (0.01 sec)

```
mysql> alter table mahasiswa type =innodb;  
Query OK, 0 rows affected, 1 warning (0.21 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

Setelah itu buat tabel dosen_pembimbing dengan (type innodb atau engine innodb) dan relasikan ke tabel mahasiswa mengambil key nim dari tabel mahasiswa sebagai foreign key di tabel dosen_pembimbing.

```
mysql> create table dosen_pembimbing(  
-> idx int(11) not null primary key auto_increment,  
-> kode_dosen varchar(5),  
-> nama_dosen varchar(20),  
-> nim varchar(20),  
-> index (nim),  
-> foreign key(nim) references mahasiswa(nim) on update cascade on delete cascade) type=innodb;  
Query OK, 0 rows affected, 1 warning (0.16 sec)  
  
mysql> desc dosen_pembimbing;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| idx   | int(11) | NO | PRI | NULL | auto_increment |  
| kode_dosen | varchar(5) | YES | | NULL | |  
| nama_dosen | varchar(20) | YES | | NULL | |  
| nim   | varchar(20) | YES | MUL | NULL | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.01 sec)
```

Nah, pada table diatas terlihat apa yang disebut dengan foreign key, yaitu nim pada table dosen_pembimbing merupakan foreign key, sedangkan pada table mahasiswa nim bertindak sebagai primary key, relasi inilah yang sering kita jumpai dalam pembuatan basis data pada sebuah sistem informasi.

References menunjukkan bahwa foreign key di suatu tabel merupakan key pada tabel lain. Pada perubahan suatu record baik itu update atau delete, dapat diikuti dengan kata : cascade, restrict atau set null. Kita bisa atur sesuai kebutuhan.

Contoh :

```
on update CASCADE on delete CASCADE
```

```
on update RESTRICT on delete SET NULL
```

Praktik

1. Membuat DataBase Perbankan dan Mengaktifkannya !
2. Membuat Tabel Bank !

Field	Type	Null	Key	Default	Extra
Kode_Bank	varchar(5)	NO	PRI		
Nama_Bank	varchar(20)	YES		NULL	
Alamat_Bank	varchar(20)	YES		NULL	

3. Mengaktifkan innodb !
4. Mengisi Data Pada Tabel Bank !

Kode_Bank	Nama_Bank	Alamat_Bank
001	Mandiri	Sempaja
002	Kaltim	Antasari
003	BRI	Pemuda
004	BCA	Sentosa

5. Membuat Tabel Nasabah ! No_Rek (Primary Key) & Kode_Bank (Foreign Key)

Field	Type	Null	Key	Default	Extra
No_Rek	varchar(12)	NO	PRI		
Kode_Bank	varchar(5)	NO	MUL		
Nama	varchar(20)	YES		NULL	
Alamat	varchar(20)	YES		NULL	
No_Telp	varchar(12)	YES		NULL	

6. Mengisi Data Pada Tabel Nasabah !

No_Rek	Kode_Bank	Nama	Alamat	No_Telp
0122331344	004	Unyil	Pahlawan	083420887612
0122364349	002	Jarjit	Sentosa	083490008876
0123012348	003	Upin	Pelita	082122448876
0123012488	003	Ipin	Kemakmuran	082133445576
0123234341	001	Ehsan	Pemuda	083422368876

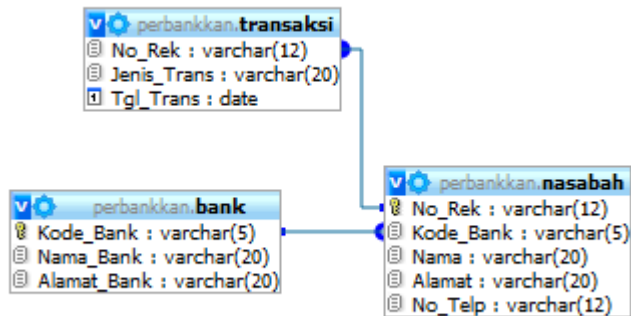
7. Membuat Tabel Nasabah ! No_ReK(Foreign Key)

Field	Type	Null	Key	Default	Extra
No_Rek	varchar(12)	NO	MUL		
Jenis_Trans	varchar(20)	YES		NULL	
Tgl_Trans	date	YES		NULL	

8. Mengisi Data Pada Tabel Transaksi !

No_Rek	Jenis_Trans	Tgl_Trans
0122364349	Transfer	2010-05-02
0122331344	Menabung	2010-08-09
0122331344	Penarikan Uang	2010-10-09
0123234341	Cek Saldo	2009-09-09
0123234341	Transfer	2011-01-01
0123012488	Menabung	2012-12-12
0123012348	Menabung	2012-12-12
0123012488	Cek Saldo	2013-12-11

9. Silahkan Liat Relasi Anda di Localhost dengan cara membuka data base Perbankan lalu klik designer di Seperti di bawah ini



10. Tampilkan Tabel Bank Dan Tabel Nasabah,

```

mysql> select * from bank;
+----+-----+-----+
| Kode_Bank | Nama_Bank | Alamat_Bank |
+----+-----+-----+
| 001       | Mandiri  | Sempaja     |
| 002       | Kaltim   | Antasari    |
| 003       | BRI      | Pemuda      |
| 004       | BCA      | Sentosa     |
+----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from Nasabah;
+----+-----+-----+-----+-----+
| No_Rek | Kode_Bank | Nama | Alamat | No_Telp |
+----+-----+-----+-----+-----+
| 0122331344 | 004 | Unyil | Pahlawan | 083420887612 |
| 0122364349 | 002 | Jarjit | Sentosa | 083490008876 |
| 0123012348 | 003 | Upin | Pelita | 082122448876 |
| 0123012488 | 003 | Ipin | Kemakmuran | 082133445576 |
| 0123234341 | 001 | Ehsan | Pemuda | 083422368876 |
+----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
  
```

11. Kemudian lakukan update data di tabel Bank dengan 1x Perintah

```
mysql> select*from bank;
```

Kode_Bank	Nama_Bank	Alamat_Bank
001	Mandiri	Sempaja
002	Kaltim	Antasari
003	BRI	Pemuda
005	BCA	Sentosa

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

```
mysql> select*from Nasabah;
```

No_Rek	Kode_Bank	Nama	Alamat	No_Telp
0122331344	005	Unyil	Pahlawan	083420887612
0122364349	002	Jarjit	Sentosa	083490008876
0123012348	003	Upin	Pelita	082122448876
0123012488	003	Ipin	Kemakmuran	082133445576
0123234341	001	Ehsan	Pemuda	083422368876

(Hasil Setelah di Update)

12. Tampilkan Tabel Nasabah Dan Tabel Transaksi

```
mysql> select*from Nasabah;
```

No_Rek	Kode_Bank	Nama	Alamat	No_Telp
0122331344	005	Unyil	Pahlawan	083420887612
0122364349	002	Jarjit	Sentosa	083490008876
0123012348	003	Upin	Pelita	082122448876
0123012488	003	Ipin	Kemakmuran	082133445576
0123234341	001	Ehsan	Pemuda	083422368876

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

```
mysql> select*from Transaksi;
```

No_Rek	Jenis_Trans	Tgl_Trans
0122364349	Transfer	2010-05-02
0122331344	Menabung	2010-08-09
0122331344	Penarikan Uang	2010-10-09
0123234341	Cek Saldo	2009-09-09
0123234341	Transfer	2011-01-01
0123012488	Menabung	2012-12-12
0123012348	Menabung	2012-12-12
0123012488	Cek Saldo	2013-12-11

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

LABORATORIUM

BAHASA PEMROGRAMAN

13. Kemudian lakukan delete data di tabel Nasabah dengan 1x Perintah

```
mysql> select*from Nasabah;
```

No_Rek	Kode_Bank	Nama	Alamat	No_Telp
0122331344	005	Unyil	Pahlawan	083420887612
0122364349	002	Jarjit	Sentosa	083490008876
0123012348	003	Upin	Pelita	082122448876
0123012488	003	Ipin	Kemakmuran	082133445576

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



```
mysql> select*from Transaksi;
```

No_Rek	Jenis_Trans	Tgl_Trans
0122364349	Transfer	2010-05-02
0122331344	Menabung	2010-08-09
0122331344	Penarikan Uang	2010-10-09
0123012488	Menabung	2012-12-12
0123012348	Menabung	2012-12-12
0123012488	Cek Saldo	2013-12-11

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

(Hasil Setelah di delete)