

PERTEMUAN 6

BASIS DATA RELASIONAL

A. TUJUAN PEMBELAJARAN

Pada pertemuan ini Mahasiswa dapat memahami suatu konsep basis data relasional, dan mampu membangun model basis data relasional serta batasan basis data relasional.

B. URAIAN MATERI

1. Basis Data Relasional

Database relasional didasarkan pada model relasional dan menggunakan kumpulan-kumpulan tabel untuk merepresentasikan data dan hubungan di antara data tersebut. ini juga termasuk DML dan DDL. Kami menyajikan pengantar yang lembut ke dasar-dasar model relasional. Kebanyakan sistem *database* relasional komersial menggunakan bahasa SQL. Dalam Bab ini kita membahas bahasa-bahasa berpengaruh lainnya.

Dr. E.F. Codd pada tahun 1970 telah memperkenalkan model relasional sistem *database* yang adalah merupakan dasar untuk sebuah Relational *Database* Management System (RDBMS). Sebelum munculnya konsep basis data relasional, dua model basis data digunakan, yaitu, basis data jaringan dan basis data hierarki. Kemudahan penggunaan dan fleksibilitas struktural RDBMS membuatnya cepat populer dan digunakan dengan banyak pemasok inovatif untuk membantu pengembangan aplikasi dan produk yang kuat yang menyediakan solusi.

Didalam *database* yang relasional, data dapat disimpan kedalam bentuk relasi dua dimensi atau tabel, dan terdapat relasi satu table dengan table lainnya. Komponen model rasional meliputi: sebuah kumpulan objek yang memiliki asosiasi atau asosiasi antar penyimpanan data, sekumpulan operator yang bisa melakukan asosiasi untuk membuat asosiasi lain, serta akurasi dan konsistensi data. Anda perlu menggunakan software RDBMS untuk membuat

suatu tabel, mengisi datanya menjadi tabel, mentransformasi data, dan menghapus data-data dari tabel. Perintah yang digunakan disebut adalah Structural Query Language (SQL), maka software RDBMS dapat digunakan untuk menjalankan perintah SQL.

2. Model Data Relasional

Perancangan basis data ialah pada pengembangan suatu sistem organisasi basis data yang pada dasarnya adalah berorientasi untuk kebutuhan para pengguna. Oleh sebab itu, pendekatan model data ini dibagi menjadi tiga tahap perancangan basis data, desain secara konseptual, desain secara logis dan secara fisik.

a. Desain Basis Data secara Konseptual

Desain *database* secara konseptual juga melibatkan penemuan dan sebuahanalisis data terhadap kebutuhan suatu organisasi. Perangkat utamanya yang dapat digunakan dalam pembuatan sebuah model data adalah diagram relasi entitas.

b. Desain Basis Data secara Logis

Desain basis data ini adalah merupakan pengembangan dari desain basis data secara konseptual memakainya ke dalam tabel-tabel. Tabel-tabel ini akhirnya dapat digunakan untuk mendeskripsikan basis data secara fisik bagi para pemakai akhir untuk pengambilan keputusan.

c. Desain Basis Data secara Fisik

Secara sistematis contohnya sebagai berikut:

- 1) Dapat menciptakan tabel yang tidak dinormalisasikan pada formular
- 2) Dapat membuat Relasi Antar Tabel
- 3) Dapat menempatkan kunci-kunci asing kedalam tabel
- 4) Dapat membuat Normalisasi Tabel
- 5) Dapat menentukan Relasi Antara Tabel-tabel
- 6) Dapat membuat Identifikasi Data

Suatu tahapan dalam normalisasi pada umumnya terdapat tiga langkah utama, yaitu sebagai berikut:

- 1) Untuk menghilangkan bentuk perulangan (redundancy)
- 2) Untuk mengubah ketergantungan yang parsial
- 3) Untuk mengubah ketergantungan yang transitif

Sebuah model relasional biasanya menggunakan kumpulan table untuk merepresentasikan data dan hubungan di antara mereka. Setiap table terdiri dari kolom, dan setiap kolom memiliki nama unik. Pada model ini basis data disusun dalam bentuk table dua dimensi. Tabel tersebut terdiri dari baris dan kolom. Perpotongan antara baris dan kolom disebut item data atau nilai data. Tabel tersebut adalah ditautkan dengan cara ini Cara menggunakan bidang kunci untuk meminimalkan duplikasi data.

Contoh Model Data Relasional:

Tabel 0.1 Tabel Model Data Relasional

NIM	NAMA	ALAMAT	DOSEN WALI
1139400015	Andi Ari	Jl. PGA No.11	WNS
1139500918	Ani Ambar	Jl. ABC No.16	TBH
1139501010	Tuti	Jl. AA No. 178	WNS

DOSEN WALI	ALAMAT
WNS	KOPO PERMAI 1
TBH	BALE ENDAH 1

Tabel 0.2 Tabel Model Data Relasional Fakultas

Fakultas	Jml_Dsn	Jml_Mhs	Jml_Jur
Teknik	152	6322	3
Ekonomi	210	4122	4
...

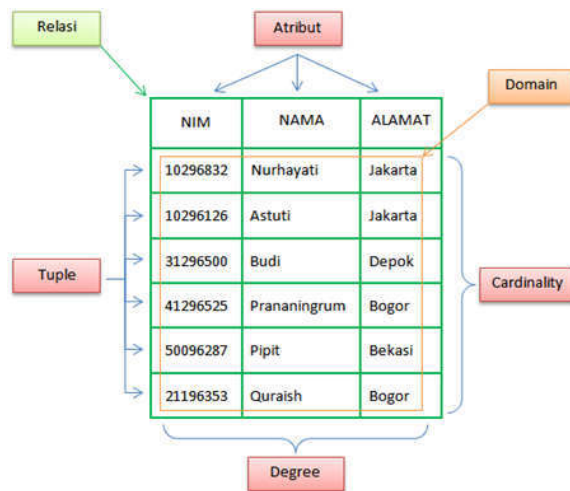
Tabel 0.3 Tabel Model Data Relasional Mahasiswa

No MK	Nama MK	SKS	NIP
KDI11892	BahasaInggris	2	12122
KAI16522	FisikaLanjut	2	23144
...

Dengan menggunakan model ini, pencarian *field* dapat dilakukan dengan cepat. Pencarian atribut terkait dalam tabel yang berbeda dapat dilakukan terlebih dahulu menggabungkan tabel dengan menggunakan atribut yang sama (operasi umum).

a. Terminologi

- Model ini menjelaskan secara logis antara data dalam *database* dengan menggambarkan kedalam bentuk tabel dua dimensi yang menunjukkan atribut.
- Istilah dalam sebuah model basis data yang relasional:

**Gambar 0.1** Contoh Relasi/Tabel Mahasiswa

- a) Relasi merupakan tabel yang memiliki baris dan kolom. Pada contoh diatas, relasi/tabel adalah mahasiswa.
- b) Atribut merupakan kolom yang terdapat dalam relasi (tabel). Pada contoh di atas, atribut dari relasi mahasiswa adalah NIM, nama, alamat.
- c) Domain merupakan kumpulan nilai yang bersifat atomic yang diizinkan (valid) sebagai nilai dalam suatu atribut. Pada contoh tabel mahasiswa di atas, domain tersebut terdiri dari atribut sebagai berikut:
 - NIM = terdiri dari kumpulan delapan digit angka yang valid (sesuai dengan aturan suatu lembaga)
 - Nama = terdiri dari kumpulan karakter string yang mempresentasikan nama orang
 - Alamat = terdiri dari kumpulan karakter string yang mempresentasikan tempat tinggal
- d) Tuple/Record merupakan kesatuan baris yang terdapat dalam relasi.
Contoh tuple pada relasi mahasiswa di atas adalah NIM = 10296832, Nama = Nurhayati, Alamat = jakarta
- e) Degree dalam suatu relasi merupakan jumlah sebuah atribut yang ada dalam sebuah relasi.
Contoh diatas, relasi memiliki 3 degree
- f) Kardinalitas dalam sebuah relasi merupakan jumlah tuple yang terdapat di dalam relasi.
Dalam contoh di atas relasi memiliki 6 kardinalitas
- g) Tabel adalah merupakan sebuah struktur penyimpanan dasar dari sebuah basis data yang relasional, yang terdiri dari satu atau lebih kolom atau lebih baris.
- h) Row adalah kombinasi dari suatu nilai kolom kedalam tabel; sebagai contohnya adalah suatu informasi departemen pada tabel Departmen, barisnya disebut "record".
- i) Column adalah menggambarkan data pada sebuah tabel; sebagai contohnya nama departemen dalam tabel Departmen.
- j) *Field* adalah pertemuan dari baris dan kolom. *Field* dapat berisi data jika pada suatu *field* tersebut tidak terdapat data didalamnya, maka *field* tersebut memiliki nilai.

- k) *Primary key* atau kunci utama yang merupakan kolom atau kumpulan suatu kolom secara unik dapat membedakan antara baris yang satu dengan baris lainnya; contoh, kode departemen. Kolom dengan kategori ini nilainya harus *unique* tidak boleh null.
- l) *Foreign key* adalah kolom atau kumpulan kolom yang mengacu ke *primary key*. *Foreign key* ini dibuat untuk memaksakan aturan-aturan sebuah relasi pada basis data. Nilai data dari *foreign key* harus sesuai pada kolom dari tabel yang diacunya atau bernilai "null".

b. Karakteristik Model Basis Data Relasional

Relasi sebuah model basis data relasional yang memiliki karakteristik sebagai berikut:

- 1) Semua elemen data pada suatu baris dan kolom tertentu harus mempunyai nilai tunggal atau nilai yang tidak dapat dibagi lagi dan bukan pengulangan.
- 2) Semua elemen data pada kolom tertentu dalam relasi yang sama harus mempunyai jenis yang sama pula.
- 3) Masing-masing kolom dalam suatu relasi harus unik.
- 4) Pada suatu relasi / tabel yang sama tidak mempunyai dua baris yang identik.

c. Komponen Relasi

Tabel sebuah relasional mempunyai 2 komponen sebagai berikut:

- 1) Intention
 - Terdiri dari dua bagian yaitu struktur penamaan dan batasan integritas.
 - a) Struktur penamaan menunjukkan nama tabel dan nama atribut lengkap dengan tipe datanya.
 - b) Batasan integritas dipengaruhi dengan integritas referensial yang meliputi key *constraint* dan referensial *constraint*.
 - c) Key *constraint* tidak boleh ada nilai null pada atribut yang digunakan sebagai PK.
 - d) Referential *constraint* memberikan aturan bahwa nilai-nilai dalam atribut kunci yang digunakan untuk menghubungkan ke basis data lain tidak boleh ada nilai null.

2) Extention

Gambaran isi dari tabel-tabel pada suatu waktu tertentu dan cenderung dapat berubah sewaktu-waktu.

d. Manfaat dan Langkah-Langkah dari Sebuah Basis Data Relasional

Berikut ini adalah langkah-langkah yang dapat memanfaatkan kembali sebuah *database* yang telah dirancang sebelumnya.

1) Relasi di pilih dari basis data

Langkah ini dapat dilakukan dengan menjaga directory atau tujuan pemakai sebagai sebuah memori bantuan di dalam relasi.

2) Menampilkan hitungan dari hitungan total

Jika sebuah sub kelompok data yang tepat sudah ditentukan dan baris relasi telah disusun secara rapi maka total atau hasil hitungan dapat dilakukan secara cepat.

3) Relasi digabungkan secara bersama-sama

Gabungan operasi yang dimaksudkan adalah untuk mengambil dua relasi dan menempatkannya secara bersama-sama agar dapat membuat relasi yang lebih besar lagi.

4) Membagi sebuah atribut yang baru

Langkah ini adalah manipulasi sebuah data ditambah beberapa parameter tambahan, jika diperlukan agar memperoleh sebuah data baru. Lalu, kolom baru dapat dibuat untuk membuat hasil relasi.

5) Mengurutkan barisnya dan memberi indeks

6) Membuat indeks merupakan penyusunan baris secara logika dalam membuat sebuah relasi menurut beberapa kunci, sedangkan pengurutan adalah penyusunan sebuah relasi secara fisiknya.

7) Menampilkan sebuah data

Langkah berikutnya dalam mendapatkan kembali data-data adalah presentasi yang dapat ditampilkan kedalam bentuk tabel, grafik, gambar dan lain sebagainya.

8) Membangun kolom dari relasi

Kegiatan ini dapat dilakukan dengan cara membangun relasi yang lebih kecil caranya dengan hanya memilih salah satu atribut yang relevan dari relasi yang ada.

9) Memilih baris dari relasi

Pemilihan suatu baris dapat membuat sebuah hubungan yang baru (lebih kecil) dengan cara mengekstraksi record yang berisi sebuah atribut dengan syarat tertentu.

3. Keuntungan dan Kelemahan model Relasional

Beberapa keuntungan dalam menggunakan model data relasional meliputi:

- a. Data dapat dengan cepat diakses
- b. Struktur basis data bisa dengan mudah dilakukan perubahan
- c. Data dipresentasikan secara logic, user tidak membutuhkan bagaimana data disimpan
- d. Mudah untuk membuat query yang kompleks dalam melakukan retrieve data
- e. Mudah untuk mengimplementasikan integritas data
- f. Data lebih akurat
- g. Mudah untuk membangun dan memodifikasi program aplikasi
- h. Telah dikembangkan Structure query language (SQL)

Kelemahan – kelemahan dari model data ini yaitu :

- a. Kelompok informasi yang berbeda harus dilakukan joined untuk melakukan retrieve data
- b. User harus familiar dengan relasi antar table
- c. User harus belajar SQL

4. Bahasa Pada Model Relasional

Bahasa yang digunakan untuk memperoleh informasi dari *database*. Terdapat dua jenis bahasa dalam relasional *database*, yaitu Bahasa formal dan bahasa komersial.

a. Bahasa Formal

Bahasa formal merupakan bahasa query yang diterjemahkan dalam bentuk simbol – simbol matematis. Contohnya kalkulus relasional dan aljabar Relasional.

a) Kalkulus Relasional

Merupakan bahasa query non-procedural di mana pemakai menspesifikasikan data apa yang dibutuhkan tanpa menspesifikasikan bagaimana untuk mendapatkannya.

Bahasa kalkulus relasional terbagi menjadi dua bagian, yaitu

- a) Kalkulus relasional tupel
- b) Kalkulus Relasional domain

b) Relational Algreba

Aljabar relasional mendefinisikan sekumpulan operasi pada relasi, paralel dengan operasi aljabar biasa seperti penjumlahan, pengurangan, atau perkalian, yang beroperasi pada bilangan. Sama seperti operasi aljabar pada bilangan yang mengambil satu atau lebih bilangan sebagai masukan dan mengembalikan bilangan sebagai keluaran, operasi aljabar relasional biasanya mengambil satu atau dua relasi sebagai masukan dan mengembalikan relasi sebagai keluaran.

Berikut ini keenam operator dalam relational aljabar :

a) Select = σ

Select digunakan untuk mendapatkan relasi baru sesuai criteria yang sudah ditentukan. Contohnya terdapat relasi mahasiswa yang disimbolkan dengan M.

Tabel 0.4 Relasi Mahasiswa untuk Contoh Proses Select

NIM	NAMA	JENIS KELAMIN	JURUSAN
0305021001	RESIKA	L	INF
0305021002	AGINDA	L	MTM
0305021003	HARINI	P	FIS
0305021004	FEBRI	P	INF

Pada operasi algebra : $\sigma_{\text{jurusan}=\text{"INF"}}(M)$ menghasilkan relasi baru sebagai berikut

Tabel 0.5 Hasil Operasi Select pada Relasi Mahasiswa

NIM	NAMA	JENIS KELAMIN	JURUSAN
0305021001	RESIKA	L	INF
0305021004	FEBRI	P	INF

b) Project = Π

Project digunakan untuk mendapatkan relasi baru yang tidak duplikat pada atribut yang sudah ditentukan. Contohnya terdapat relasi mahasiswa yang disimpulkan dengan M.

Tabel 0.6 Relasi Mahasiswa untuk Contoh Operasi Project

NIM	NAMA	JENIS KELAMIN	JURUSAN
0305021001	RESIKA	L	INF
0305021002	AGINDA	L	MTM
0305021003	HARINI	P	INF
0305021004	FEBRI	P	INF

Pada operasi algebra : Π (jenis kelamin, jurusan) (M) menghasilkan relasi sebagai berikut

Tabel 0.7 Hasil Operasi Project pada Relasi Mahasiswa

JENIS KELAMIN	JURUSAN
L	INF
L	MTM
P	INF

c) Union : U

Union berfungsi untuk menggabungkan dua relasi sehingga menghasilkan relasi baru hasil penggabungan yang tidak duplikat.

Mahasiswa (M1)

Mahasiswa (M2)

Tabel 0.8 Tabel Mahasiswa

NIM	NAMA
0305021001	RESIKA
0305021002	AGINDA
0305021003	HARINI
0305021004	FEBRI

NIM	NAMA
0305021010	Ani
0305021011	Budi
0305021004	Febri

Pada operasi algebra : $M1 \cup M2$ akan menghasilkan relasi baru sebagai berikut.

Tabel 0.9 Hasil Operasi Union pada Relasi Mahasiswa 1 dan Mahasiswa 2

NIM	NAMA
0305021001	RESIKA
0305021002	AGINDA
0305021003	HARINI
0305021004	FEBRI
0305021010	ANI
0305021011	BUDI

d) Set Difference : -

Set difference berfungsi untuk menghasilkan perbedaan antar dua relasi.

Contohnya terdapat relasi M1 dan M2

Mahasiswa (M1)

Mahasiswa (M2)

Tabel 0.10 Tabel Mahasiswa

NIM	NAMA
0305021001	RESIKA
0305021002	AGINDA
0305021003	HARINI
0305021004	FEBRI

NIM	NAMA
0305021010	Ani
0305021011	Budi
0305021004	Febri

Pada operasi algebra $M1 - M2$ maka akan menghasilkan relasi sebagai berikut :

Tabel 0.11 Hasil Operasi Difference antara Relasi Mahasiswa 1 dan Mahasiswa 2

NIM	NAMA
0305021001	RESIKA
0305021002	AGINDA
0305021003	HARINI

e) Cartesian product : x

Cartesian produk digunakan untuk menggabungkan dua relasi
(Pada atribut yang berbeda)

Mahasiswa (M1)

Mahasiswa (M2)

Tabel 0.12 Tabel mahasiswa

NIM	NAMA
0305021001	RESIKA
0305021004	FEBRI

JENIS KELAMIN	JURUSAN
L	INF
P	INF

Pada operasi algebra $M1 \times M2$ akan menghasilkan relasi sebagai berikut.

Tabel 0.13 Hasil Operasi Kartesian Produk antara M1 dan M2

NIM	NAMA	JENIS KELAMIN	JURUSAN
0305021001	RESIKA	L	INF
0305021001	RESIKA	P	INF
0305021004	FEBRI	L	INF
0305021004	FEBRI	P	INF

f) Rename : p

Rename p berfungsi untuk mengubah nama relasi menjadi nama yang baru. Contoh :

$p_{x1}(R1)$ = X1 merupakan nama baru untuk hasil dari R1

b. Bahasa Komersial

Bahasa Komersial atau Bahasa query yang dirancang sendiri oleh programmer menjadi sebuah program atau aplikasi agar pengguna lebih mudah memakainya (user friendly).

Contohnya sebagai berikut:

- 1) QUEL sumber pada bahasa kalkulus relasional
- 2) QBE sumber pada bahasa kalkulus relasional
- 3) SQL sumber pada bahasa kalkulus relasional dan aljabar relasional.

C. SOAL LATIHAN/TUGAS

1. Sebutkan dan jelaskan basis data relasional ?
2. Sebutkan dan jelaskan Istilah dari basis data relasional ?
3. Perhatikan rancangan basis data berikut

Tabel produk

Kode produk	Nama produk	Harga	Kode supplier
0001	Hardisk	500000	SP001
0002	Memory	200000	SP001
0003	Monitor	850000	SP002
0004	Power supply	125000	SP002

Tentukan atribut, *domain*, *Tuple*, *Degree*, dan kardinalitas pada tabel di atas ?

4. Sebutkan keuntungan Model relasional ?
5. Sebutkan Kelemahan Model relasional ?
6. Sebutkan dan jelaskan bahasa pada model relasional ?

D. REFERENSI

Elmasri, R. & Navathe, S.B., 2011. *Fundamentals of Database System Sixth Edition*. Boston: Addison-Wesley.

Dantes, Gede Rasben, dkk., 2019. *Pengantar Basis data*. Depok : Rajawali Pers.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *Database System Concept ; Sixth Edition*. New York: McGraw-Hill.

GLOSARIUM

Relasi merupakan sebuah tabel yang memiliki baris dan kolom.

Atribut merupakan kolom yang terdapat dalam relasi (tabel).

Domain merupakan kumpulan nilai yang bersifat atomic yang diizinkan (valid) sebagai nilai dalam suatu atribut.

Tuple/Record merupakan kesatuan baris yang terdapat dalam relasi.

Degree merupakan jumlah atribut yang ada dalam sebuah relasi.

Kardinalitas merupakan jumlah *tuple* yang terdapat di dalam relasi.

Select digunakan untuk mendapatkan relasi baru sesuai criteria yang sudah ditentukan.

Project digunakan untuk mendapatkan relasi baru yang tidak duplikat pada atribut yang sudah ditentukan.

Union berfungsi untuk menggabungkan dua relasi sehingga menghasilkan relasi baru hasil penggabungan yang tidak duplikat.

Set Difference berfungsi untuk menghasilkan perbedaan antar dua relasi.

Cartesian product digunakan untuk menggabungkan dua relasi (Pada atribut yang berbeda).

Rename berfungsi untuk mengubah nama relasi menjadi nama yang baru