

# NORMALISASI DATA

# Perancangan Basis Data

- ▶ Tujuan dilakukan perancangan suatu basis data yaitu supaya kita bisa memiliki basis data yang kompak dan efisien dalam penggunaan ruang penyimpanan, cepat dalam pengaksesan dan mudah dalam pemanipulasian (tambah, ubah, hapus) data.
- ▶ Dalam merancang basis data, kita dapat melakukannya dengan :
  - a) Menerapkan Normalisasi terhadap struktur table yang telah diketahui
  - b) Langsung membuat model Entity – Relationship.

# Definisi Normalisasi

- ▶ Normalisasi merupakan teknik analisis data yang mengorganisasikan atribut-atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang non-redundant, stabil dan fleksibel.
- ▶ Normalisasi dilakukan sebagai uji coba pada suatu relasi secara berkelanjutan untuk menentukan apakah relasi itu sudah baik, yaitu dapat dilakukan proses insert, update, delete dan modifikasi pada satu atau beberapa atribut tanpa mempengaruhi integritas data dalam relasi tersebut.

# Normalisasi VS Model ER

- a. Normalisasi merupakan pendekatan dalam membangun desain logik basis data relasional yang tidak secara langsung berkaitan dengan model data, tetapi dengan menerapkan sejumlah aturan dan kriteria standar untuk menghasilkan struktur table yang normal.
- b. Pembuatan model ER merupakan Pendekatan dalam membangun desain basis data, dimana kelompok – kelompok data dan relasi antar kelompok data tersebut diwujudkan dalam bentuk diagram.

# Normalisasi VS Model ER

- a) Kedua pendekatan ini sering dilakukan bersama-sama, berganti-ganti.
- b) Dari Fakta yang kita miliki, dilakukan normalisasi
- c) Untuk kepentingan evaluasi dan dokumentasi, hasil normalisasi itu diwujudkan dalam model data.
- d) Model data yang sudah jadi bisa dimodifikasi dengan pertimbangan tertentu
- e) Hasil modifikasi, diimplementasikan dalam bentuk sejumlah struktur table dalam sebuah basis data
- f) Struktur ini bisa diuji kembali dengan menerapkan aturan-aturan normalisasi.

# Istilah-istilah dalam Normalisasi

- ▶ Atribut
- ▶ Key
- ▶ Tipe dan Domain
- ▶ Ketergantungan

# Atribut Tabel

- ▶ Atribut identik dengan pemakaian istilah kolom data.
- ▶ Fungsinya yaitu untuk menunjukkan fungsinya sebagai pembentuk karakteristik (sifat-sifat) yang melekat pada sebuah table.
- ▶ Penerapan aturan-aturan normalisasi terhadap atribut – atribut pada sebuah table bisa berdampak pada penghilangan kolom tertentu, penambahan kolom baru atau penambahan table baru.

# Key dan Atribut Deskriptif

- ▶ Key adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (row) dalam table secara unik.
- ▶ Jika suatu atribut dijadikan sebagai key, maka tidak boleh ada dua atau lebih baris data dengan nilai yang sama untuk atribut tersebut
- ▶ Atribut deskriptif adalah atribut – atribut yang tidak menjadi atau merupakan anggota dari Primary Key.



# Macam – macam Key

- ▶ *Superkey*
- ▶ *Candidate Key*
- ▶ *Primary Key*

# Superkey

- ▶ Satu atau lebih atribut yang dapat membedakan setiap baris data dalam sebuah table secara unik.
- ▶ Bisa terjadi, ada lebih dari 1 kumpulan atribut yang bersifat seperti itu pada sebuah table
- ▶ Contoh :
  - ▶ {nim, nama\_mhs, alamat\_mhs, tgl\_lahir}
  - ▶ {nim, nama\_mhs, alamat\_mhs}
  - ▶ {nim, nama\_mhs}
  - ▶ {nama\_mhs}
  - ▶ {nim}

# Candidate Key

- ▶ Kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah table secara unik
- ▶ Sebuah Candidate-Key tidak boleh berisi kumpulan atribut yang telah menjadi Superkey yang lain.
- ▶ Candidate Key pasti Super Key, tapi belum tentu sebaliknya.
- ▶ Contoh :
  - ▶ {nim}
  - ▶ {nama\_mhs}
- ▶ Dalam sebuah table dimungkinkan ada lebih dari satu Candidate key
- ▶ Salah satu Candidate Key, dapat dijadikan Sebagai Primary Key.

# Atribut sederhana (Simple Attribute) & Atribut Komposit (Composite Attribute)

- ▶ Atribut sederhana adalah atribut yang tidak dapat dipilah lagi
- ▶ Atribut komposit merupakan atribut yang masih dapat diuraikan lagi menjadi sub-sub atribut yang masing-masing memiliki makna.

## Atribut Tunggal (*Single-Valued Attribute*) & Atribut bernilai banyak (*Multivalued Attribute*)

- ▶ Atribut nilai tunggal ditunjukkan pada atribut – atribut yang memiliki paling banyak satu nilai untuk setiap baris data.
- ▶ Sedangkan atribut bernilai banyak ditunjukkan pada atribut-atribut yang dapat diisi lebih dari 1 nilai.

# Atribut harus Bernilai (Mandatory Attribute) & Nilai Null

- ▶ Mandatory Attribute yaitu sejumlah atribut pada sebuah table yang harus berisi data. Jadi nilainya tidak boleh kosong.
- ▶ Nilai Null digunakan untuk mengisi atribut-atribut yang nilainya memang belum siap atau tidak ada.

# Atribut Turunan (Derived Attribute)

- ▶ Atribut turunan adalah atribut yang nilai-nilainya diperoleh dari pengolahan data atau dapat diturunkan dari atribut atau table lain yang berhubungan.
- ▶ Atribut ini dapat diiadakan dari sebuah table, karena nilai-nilainya bergantung pada nilai yang ada di atribut lainnya.

# Domain dan Tipe Data

- ▶ Penetapan tipe data akan berimplikasi pada adanya batas-batas nilai yang mungkin disimpan / diisikan ke setiap atribut (kolom) tersebut.
- ▶ Perbedaan Tipe Data dengan domain yaitu Tipe data lebih merujuk pada kemampuan penyimpanan data yang mungkin bagi suatu atribut secara fisik, tanpa melihat layak/tidaknya data tersebut bila dilihat dari kenyataan pemakaian, Sementara domain lebih ditekankan pada batas-batas nilai yang diperbolehkan bagi suatu atribut, dilihat dari kenyataan yang ada.



# ANOMALI

- ▶ Anomali adalah proses basis data yang memberikan efek samping yang tidak diharapkan (misalnya menyebabkan tidak konsistennya data atau membuat suatu data menjadi hilang ketika data dihapus)
- ▶ Macam – macam anomaly :
  - ▶ Anomali peremajaan
  - ▶ Anomali penghapusan
  - ▶ Anomali Penyisipan

# Anomali Penyisipan (Insertion Anomaly)

- ▶ Anomali Penyisipan yaitu error atau kesalahan yang terjadi sebagai akibat operasi menyisipkan tuple/record pada sebuah relasi.
- ▶ Contoh :

No Pasien	Kode Obat	Harga Obat
P001	Kd 01	2000
P002	Kd 02	4500
P003	Kd 03	2000

- ▶ Jika ada obat baru yang akan dimasukkan/disisipkan, maka obat tersebut tidak dapat disisipkan ke dalam table relasi sampai ada pasien yang mengambil jenis obat tersebut

# Anomali Penghapusan (*Deletion Anomaly*)

- ▶ Error atau kesalahan yang terjadi sebagai akibat operasi penghapusan terhadap tuple/record dari sebuah relasi.
- ▶ Contoh :  
Jika Pasien yang memiliki No\_Pasien P001 membatalkan/ tidak jadi menebus resep obat tersebut, maka record tersebut dihapus, akan menyebabkan hilangnya informasi tentang Kode\_Obat Kd 01.

# Anomali Peremajaan

- ▶ Anomali ini terjadi bila ada perubahan pada sejumlah data yang mubazir, tetapi tidak seluruhnya diubah. Error atau kesalahan yang terjadi sebagai akibat operasi perubahan tuple/record dari sebuah relasi
- ▶ Contoh :
- ▶ Jika harga obat untuk kode\_obat Kd 01 dinaikkan menjadi 5000, maka harus dilakukan update data di table utama Obat, untuk menjaga data agar selalu konsisten

# Dependensi

- ▶ Dependensi merupakan konsep dasar normalisasi yang menjelaskan hubungan antar atribut, atau lebih khusus menjelaskan nilai suatu atribut yang menentukan nilai atribut lainnya.
- ▶ Fungsinya menjadi acuan dekomposisi data ke dalam bentuk yang lebih efisien.
- ▶ Macam-macam dependensi :
  - ▶ Dependensi Fungsional
  - ▶ Dependensi Fungsional Penuh
  - ▶ Dependensi Total
  - ▶ Dependensi Transitif

# Dependensi Fungsional

- ▶ Ketergantungan Fungsional (*Functional Dependency*) sering disingkat KF.
- ▶ Jika sebuah table T berisi minimal 2 buah atribut yaitu A dan B, maka dapat dinyatakan dengan notasi :  $A \rightarrow B$  , yang berarti A secara fungsional menentukan B atau B secara fungsional bergantung pada A, jika dan hanya jika setiap kumpulan baris data(row) yang ada di table T, pasti ada dua baris data di table T dengan nilai A yang sama, maka nilai B pasti juga sama.
- ▶ Definisi formal :
  - ▶ 2 baris data (row) R1 dan R2 dalam table T
  - ▶ Dimana  $A \rightarrow B$ , jika  $R1(A) = R2(A)$ , maka  $R1(B) = R2(B)$

# Contoh Dependensi Fungsional

Mata_kuliah	NIM	Nama_mhs	index
Struktur Data	10215111	Ujang	A
Basis Data	10215333	Asep	C
Bahasa Indonesia	10215333	Asep	A
Bahasa Indonesia	10215876	Cecep	B

- ▶ NIM → Nama\_mhs
- ▶ {Mata\_kuliah, NIM} → index
- ▶ Keterangan :
  - ▶ Bagian yang terletak di sebelah kiri panah biasa disebut penentu (determinan) dan bagian yang terletak di sebelah kanan panah disebut yang tergantung (dependen)
  - ▶ Tanda {} biasa digunakan kalau ada lebih dari satu atribut, baik pada penentu maupun yang tergantung

# Dependensi Fungsional Penuh

- ▶ Definisi dari dependensi fungsional penuh adalah suatu atribut Y mempunyai dependensi fungsional penuh terhadap atribut X jika :
  - ▶ Y mempunyai dependensi fungsional terhadap X
  - ▶ Y tidak memiliki dependensi terhadap bagian dari X



# Contoh Dependensi Fungsional Penuh

- ▶ Pelanggan (id\_pelanggan, nama\_pelanggan, kota, no\_telp)
- ▶ Maka Dependensi Fungsional Penuh :
  - a. {Kode\_Pelanggan, kota}  $\rightarrow$  no\_telp
  - b. Kode\_Pelanggan  $\rightarrow$  no\_telp

Kondisi 1 : no\_telp bergantung pada {Kode\_Pelanggan, kota} dan juga pada

Kondisi 2 : no\_telp bergantung pada kode\_pelanggan, yang merupakan bagian dari kondisi 1, maka no\_telp tidak mempunyai dependensi fungsional sepenuhnya terhadap {Kode\_Pelanggan, Kota}, Dengan kata lain no\_telp hanya mempunyai dependensi penuh terhadap Kode\_Pelanggan.

# Dependensi Total

- ▶ Definisi dari dependensi total adalah suatu atribut Y mempunyai dependensi total terhadap atribut X, jika
  - ▶ Y mempunyai dependensi fungsional terhadap X
  - ▶ X memiliki dependensi fungsional terhadap Y

Dinyatakan dengan notasi  $X \leftrightarrow Y$

# Contoh Dependensi Total

Kode_Pemasok	Nama_Pemasok	Kota
K1	Kartika	Jakarta
C1	Citra	Bandung
C2	Candra	Jakarta

Kode\_Pemasok  $\leftrightarrow$  Nama\_Pemasok, karena setiap kode tidak mempunyai nama yang sama

# Dependensi Transitif

- ▶ Definisi dari dependensi transitif yaitu suatu atribut Z mempunyai dependensi transitif terhadap atribut X jika :
  - ▶ Y mempunyai dependensi fungsional terhadap X
  - ▶ Z juga memiliki dependensi fungsional terhadap bagian dari Y

# Contoh Dependensi Transitif

Mata_Kuliah	Waktu	Ruang	Gedung
Struktur Data	Senin, 08.00 – 10.00	Ruang A	Baru
	Rabu, 08.00 – 11.15	LAB 1	Lama
Statistik Lanjutan	Selasa, 09.00 – 11.00	Ruang A	Baru
Pengantar Basis Data	Rabu, 11.30 – 13.00	Ruang B	Baru

- ▶ Maka dependensi transitifnya adalah :
- ▶ Mata\_Kuliah → {Ruang, Waktu}
- ▶ Ruang → Gedung
- ▶ Terlihat bahwa : Mata\_Kuliah → Ruang → Gedung
- ▶ Gedung memiliki dependensi transitif terhadap atribut Mata\_Kuliah.



SELESAI

TERIMA KASIH