

# TEKNIK NORMALISASI 1

## Beberapa pengertian normalisasi :

Normalisasi merupakan proses pengelompokan elemen data menjadi tabel-tabel yang menunjukkan entity dan relasinya.

Normalisasi adalah proses pengelompokan atribut-atribut dari suatu relasi sehingga membentuk Well Structure Relation.

## WELL STRUKTURE RELATION

adalah sebuah relation yang jumlah kerangkapan datanya sedikit (minimum *Amount Of Redundancy*), serta memberikan kemungkinan bagi user untuk melakukan INSERT, DELETE, dan MODIFY terhadap baris-baris data pada relation tersebut, yang tidak berakibat terjadinya ERROR atau INKONSESTENSI DATA, yang disebabkan oleh operasi - operasi tersebut.

Contoh : Terdapat sebuah relation Course, dengan ketentuan sbb:

- o Setiap mahasiswa hanya boleh mengambil satu matakuliah saja.
- o Setiap matakuliah mempunyai uang kuliah yang standar (tidak tergantung pada mahasiswa yang mengambil matakuliah tsb).

## RELATION COURSE

STUDENT-ID	KODE-MTK	BIAYA
92130	CS-200	75
92200	CS-300	100
99250	CS-300	75
92425	CS-400	150
92500	CS-300	100
92575	CD-500	50

Relation di atas merupakan sebuah relation yang sederhana dan terdiri dari 3 kolom/atribut. Bila diteliti secara seksama, maka akan ditemukan redundancy pada datanya, dimana biaya kuliah selalu berulang pada setiap mhs. Akibatnya besar kemungkinan terjadi Error atau inkonsistensi data, bila dilakukan update terhadap relation tsb yang disebut dengan Anomali.

## ANOMALI

merupakan penyimpangan-penyimpangan atau Error atau Inkonsistensi data yang terjadi pada saat dilakukan proses insert, delete ataupun modify.

Terdapat 3 jenis Anomali :

### 1. Insertion Anomali

Error yang terjadi sebagai akibat operasi insert record/tuple pada sebuah relation.

Contoh: Ada matakuliah baru (CS-600) yang akan diajarkan, maka matakuliah tsb tidak bisa di insert ke dalam relation tsb sampai ada mhs yang mengambil matakuliah tsb.

### 2. Deletion Anomali

Error yang terjadi sebagai akibat operasi delete record/tuple pada sebuah relation.

Contoh: Mhs dengan student-id 92-425, memutuskan untuk batal ikut kuliah CS-400, karena dia merupakan satu-satunya peserta matakuliah tsb, maka bila record/tuple tsb didelete akan berakibat hilangnya informasi bahwa mata-kuliah CS-400, biayanya 150.

### 3. Update Anomali

Error yang terjadi sebagai akibat inkonsistensi data yang terjadi sebagai akibat dari operasi update record/tuple dari sebuah relation.

Contoh: Bila biaya kuliah untuk matakuliah CS-200 dinaikan dari 75 menjadi 100, maka harus dilakukan beberapa kali modifikasi terhadap record-record, tuple-tuple mhs yang mengambil matakuliah CS-200, agar data tetap konsisten.

Berdasarkan teori normalisasi, relation course dipecah menjadi 2 relation terpisah, sbb

STUDENT-ID	KODE-MTK	KODE-MTK	BIAYA
92130	CS-200	CS-200	75
92200	CS-300	CS-300	100
92250	CS-200	CS-400	150
92425	CS-400	CS-500	50
92500	CS-300		
92575	CD-500		

#### Problem-Problem Pada Relation Yang Sudah Dinormalisasi

- o Performance problem
- o Referential Integrity Problem

#### Beberapa Konsep Yang Harus Diketahui

1. Field/Atribut Kunci
2. Kebergantungan Fungsi

#### >> Field/Atribut Dalam Database

1. *Super key*, himpunan dari satu atau lebih entitas yang digunakan untuk mengidentifikasi secara unik sebuah entitas dalam entitas set.
2. *Candidate key*, satu attribute atau satu set minimal attribute yang mengidentifikasi secara unik suatu kejadian yang spesifik dari entity.

#### SALES

S#	SNAME	KODE
S1	ADI	1002
S2	RAFI	1001
S3	HANY	1003

#### PESANAN

KODE	P#
1002	2648
1001	2649
1003	2641

- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>• Super key = S#, SNAME, KODE</li> <li>• Candidat key = S#, SNAME</li> <li>• Primary key = S#</li> <li>• Alternatif key = SNAME</li> <li>• Foreign key = KODE</li> </ul> | <ul style="list-style-type: none"> <li>• Super key = S#, SNAME, KODE</li> <li>• Candidat key = S#, SNAME</li> <li>• Primary key = S#</li> <li>• Alternatif key = SNAME</li> <li>• Foreign key = KODE</li> </ul> |
|---|---|

## >> Ketergantungan Kunci

### 1. Ketergantungan Fungsional (Fungsional Dependent)

Keterkaitan antar hubungan antara 2 atribut pada sebuah relasi. Dituliskan dengan cara :  $A \rightarrow B$ , yang berarti: Atribut B fungsionalitas Dependent terhadap atribut A atau Isi (*value*) atribut A menentukan isi atribut B.

Definisi dari functional dependent : Diketahui sebuah relasi R, atribut Y dari R adalah FD pada atribut X dari R ditulis  $R.X \rightarrow R.Y$  jika dan hanya jika tiap harga X dalam R bersesuaian dengan tepat satu harga Y dalam R.

### 2. Fully Functionally Dependent (FFD)

Suatu rinci data dikatakan fully functional dependent pada suatu kombinasi rinci data jika functional dependent pada kombinasi rinci data dan tidak functional dependent pada bagian lain dari kombinasi rinci data.

Definisi dari FFD: Atribut Y pada relasi R adalah FFD pada atribut X pada relasi R jika Y FD pada X tidak FD pada himpunan bagian dari X.

### 3. Ketergantungan Partial

Sebagian dari kunci dapat digunakan sebagai kunci utama.

### 4. Ketergantungan Transitif

Menjadi atribut biasa pada suatu relasi tetapi menjadi kunci pada relasi lain.

### 5. Determinan

Suatu atribut (field) atau gabungan atribut dimana beberapa atribut lain bergantung sepenuhnya pada atribut tersebut.