

# Pertemuan 5

# Normalization

Nur Iksan

Pendidikan Teknik Informatika dan Komputer UNNES

# normalisasi

**Normalisasi** : Teknik/pendekatan yang digunakan dalam membangun disain logik database relasional melalui organisasi himpunan data dengan tingkat ketergantungan fungsional dan keterkaitan yang tinggi sedemikian sehingga menghasilkan struktur tabel yang normal.

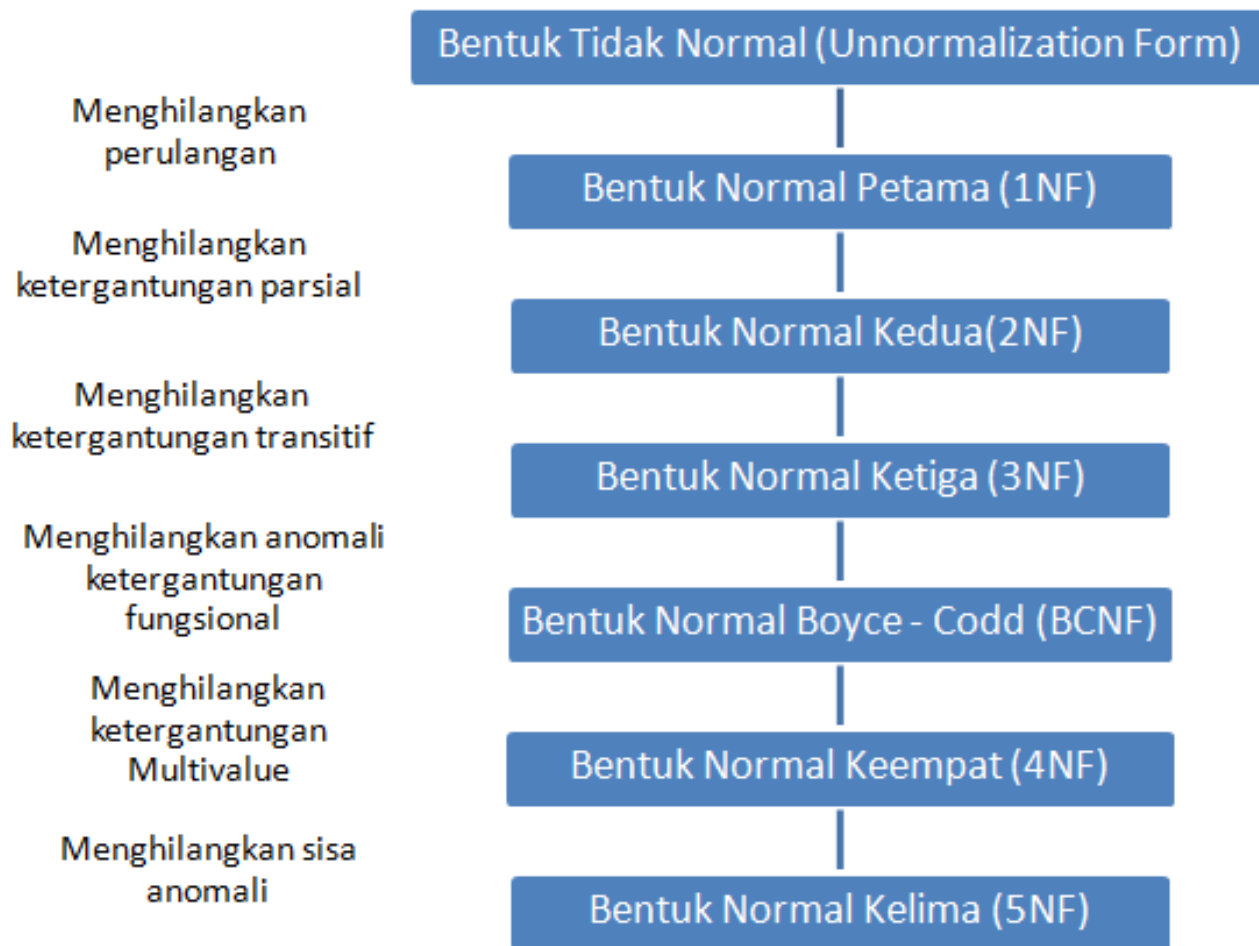
## Tujuan :

- **Minimalisasi redundansi (pengulangan data)**
- **Memudahkan identifikasi entitas**
- **Mencegah terjadinya anomali**

## Beberapa bentuk normal (normal forms, NF) :

- **1NF, 2NF, 3NF, BCNF**  
based on keys and functional dependencies
- **4NF, 5NF**  
based on keys and multi-valued dependencies

# Tahapan Proses Normalisasi



## First Normal Form (1NF)

- Suatu relasi disebut memenuhi bentuk normal pertama (1NF) jika dan hanya jika setiap atribut dari relasi tersebut hanya memiliki nilai tunggal dan tidak ada pengulangan grup atribut dalam baris.
- Bentuk 1NF tidak boleh mengandung grup atribut yang berulang.
- Menentukan kunci primer/gabungan atribut untuk membedakan antar baris
- Tujuan membentuk 1NF :
  - semantik tabel menjadi lebih eksplisit (say anything once).
  - semua operator aljabar relasional dapat diaplikasikan pada tabel.

## First Normal Form (1NF)

**Tabel : Sales**

| IDSales | NamaSales | Telepon             |
|---------|-----------|---------------------|
| ADN006  | Yeni, SE  | 3517261, 3520165    |
| ADN007  | Memey     | 4744621,08122861427 |
| ADN008  | Tina      | 08566241521         |
| ADN009  | Ir. Yanto | 7265122, 7123910    |
| ADN010  | Made      | 6723192             |

**non-atomic**

**Unnormalized  
Not 1NF**

**1NF**

| IDSales | NamaSales | Telepon     |
|---------|-----------|-------------|
| ADN006  | Yeni, SE  | 3517261     |
| ADN006  | Yeni, SE  | 3520165     |
| ADN007  | Memey     | 4744621     |
| ADN007  | Memey     | 08122861427 |
| ADN008  | Tina      | 08566241521 |
| ADN009  | Ir. Yanto | 7265122     |
| ADN009  | Ir. Yanto | 7123910     |
| ADN010  | Made      | 6723192     |

## First Normal Form (1NF)

**Unnormalized  
Not 1NF**

**Tabel : Buku**

**repeated**

| ISBN          | Thn_Terbit | ID_Pengarang | Nama_Pengarang | ID_Pengarang | Nama_Pengarang |
|---------------|------------|--------------|----------------|--------------|----------------|
| 12-1202-19222 | 1992       | K0121        | Aris M         | K1021        | Kosim P        |
| 11-1090-29101 | 2001       | K1021        | Kosim P        |              |                |
| 11-1090-29102 | 2001       | K2091        | K Odelia       | K0121        | Aris M         |
| 12-1201-90871 | 2002       | K2092        | Renaldi        | K2091        | K Odelia       |
| 13-2089-12910 | 2001       | K2019        | Samsuri J      |              |                |

| ISBN          | Thn_Terbit | ID_Pengarang | Nama_Pengarang |
|---------------|------------|--------------|----------------|
| 12-1202-19222 | 1992       | K0121        | Aris M         |
| 12-1202-19222 | 1992       | K1021        | Kosim P        |
| 11-1090-29101 | 2001       | K1021        | Kosim P        |
| 11-1090-29102 | 2001       | K2091        | K Odelia       |
| 11-1090-29102 | 2001       | K0121        | Aris M         |
| 12-1201-90871 | 2002       | K2092        | Renaldi        |
| 12-1201-90871 | 2002       | K2091        | K Odelia       |
| 13-2089-12910 | 2001       | K2019        | Samsuri J      |

**1NF**

## Second Normal Form (2NF)

- **Suatu relasi disebut memenuhi bentuk normal kedua (2NF) jika dan hanya jika :**
  - 1. memenuhi 1NF**
  - 2. setiap atribut yang bukan kunci utama tergantung secara fungsional terhadap semua atribut kunci dan bukan hanya sebagian atribut kunci (fully functionally dependent).**
- **Untuk normalisasi ke bentuk 2NF, maka tabel 1NF didekomposisi menjadi beberapa tabel yang masing-masing memenuhi 2NF.**
- **Bila terdapat ketergantungan parsial maka : eliminate.**
- **Tujuan membentuk 2NF :**
  - :: semantik tabel 2NF menjadi lebih eksplisit (fully FD)**
  - :: mengurangi update anomali yang masih mungkin terjadi pada 1NF**

## Second Normal Form (2NF)

**Contoh :**

Diketahui tabel  $R=(\underline{A},\underline{B},C,D,E)$  ; A,B kunci utama (primary key)  
dengan FD :  $A,B \rightarrow C,D,E$  maka tabel R memenuhi 2NF sebab :

$A,B \rightarrow C,D,E$  berarti :

$A,B \rightarrow C,$

$A,B \rightarrow D$  dan

$A,B \rightarrow E$

Jadi semua atribut bukan kunci utama tergantung penuh pada (A,B).



## Second Normal Form (2NF)

**Bagaimana bila  $R = (\underline{A}, \underline{B}, C, D, E)$  tetapi dengan FD :  $(A, B) \rightarrow (C, D)$  dan  $B \rightarrow E$ . Apakah memenuhi 2NF ?**

**Jelas bahwa R bukan 2NF karena ada atribut E yang bergantung hanya pada atribut B saja dan bukan terhadap (A,B).**

**Dari FD :  $(A, B) \rightarrow (C, D)$  juga mencerminkan bahwa hanya C dan D saja yang bergantung secara fungsional terhadap (A,B), tidak untuk E. Jadi bukan 2NF.**

**Untuk mengubah menjadi 2NF, lakukan dekomposisi menjadi :  $R1 = (\underline{A}, \underline{B}, C, D)$  dan  $R2 = (\underline{B}, E)$ . Tampak R1 dan R2 memenuhi 2NF.**

## Second Normal Form (2NF)

Diketahui Workshop = (NIM,Modul,Biaya,Grade)

### Peserta Workshop

| <u>NIM</u> | <u>Modul</u> | Biaya | Grade |
|------------|--------------|-------|-------|
|------------|--------------|-------|-------|

Key : NIM+Modul  
FD : Modul → Biaya

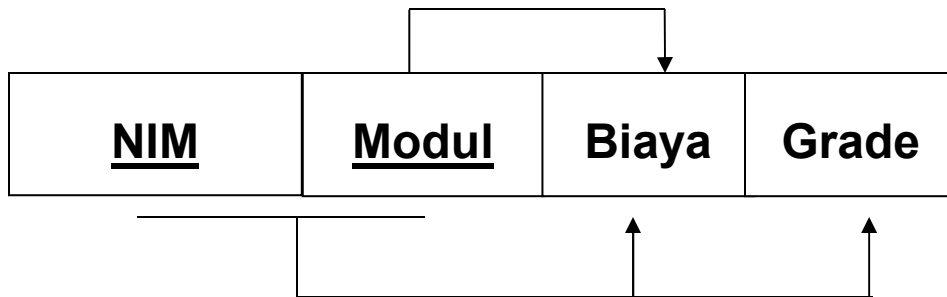
(Biaya ditentukan oleh Modul yang diambil mahasiswa)

### Tabel biaya peserta workshop

| <u>NIM</u>    | <u>Modul</u> | Biaya  | Grade |
|---------------|--------------|--------|-------|
| P11.2004.0129 | VB.Net       | 250000 | A     |
| P11.2004.0130 | Prolog       | 100000 | A     |
| P11.2004.0129 | Prolog       | 100000 | B     |
| P11.2004.0201 | Delphi 6     | 150000 | A     |
| P11.2004.0250 | VB.Net       | 250000 | B     |

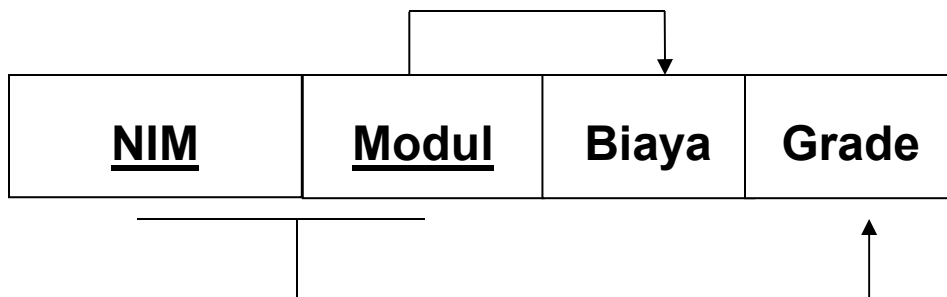
- 1NF
- Not 2NF  
Sebab dalam tabel ini, Biaya tidak bergantung penuh pada atribut kunci (NIM,Modul)

## Second Normal Form (2NF)



(NIM,Modul) = key  
(Modul) → Biaya (partial)  
(NIM,Modul) → Grade (full)

← **Eliminate**



**Make Decomposition :**

**Works1 = (NIM,Modul,Grade)**

**Works2 = (Modul,Biaya)**

**Fully Dependency**

## Second Normal Form (2NF)

### Workshop

| <u>NIM</u>    | <u>Modul</u> | Biaya  | Grade |
|---------------|--------------|--------|-------|
| P11.2004.0129 | VB.Net       | 250000 | A     |
| P11.2004.0130 | Prolog       | 100000 | A     |
| P11.2004.0129 | Prolog       | 100000 | B     |
| P11.2004.0201 | Delphi 6     | 150000 | A     |
| P11.2004.0250 | VB.Net       | 250000 | B     |

**Better Than  
1NF**

| <u>NIM</u>    | <u>Modul</u> | Grade |
|---------------|--------------|-------|
| P11.2004.0129 | VB.Net       | A     |
| P11.2004.0130 | Prolog       | A     |
| P11.2004.0129 | Prolog       | B     |
| P11.2004.0201 | Delphi 6     | A     |
| P11.2004.0250 | VB.Net       | B     |

**Works1**

| <u>Modul</u> | Biaya  |
|--------------|--------|
| VB.Net       | 250000 |
| Prolog       | 100000 |
| Delphi 6     | 150000 |

**Works2**

## Third Normal Form (3NF)

- Suatu relasi disebut memenuhi bentuk normal ketiga (3NF) jika dan hanya jika :
  1. memenuhi 2NF
  2. setiap atribut yang bukan kunci tidak tergantung secara fungsional terhadap atribut bukan kunci yang lain dalam relasi tsb (tidak terdapat ketergantungan transitif pada atribut bukan kunci).

### Another Definition :

- Suatu relasi disebut memenuhi bentuk normal ketiga (3NF) jika dan hanya jika setiap FD nontrivial :  $X \rightarrow A$ , dimana X dan A atribut (atau kompositnya), memenuhi salah satu kondisi :
  1. X adalah superkey atau
  2. A merupakan anggota candidate key

## Third Normal Form (3NF)

- Jika suatu relasi sudah memenuhi 2NF tapi tidak memenuhi 3 NF, maka untuk normalisasi ke bentuk 3NF, tabel 2NF didekomposisi menjadi beberapa tabel hingga masing-masing memenuhi 3NF.
- Tujuan membentuk 3NF :
  - :: semantik tabel 3NF menjadi lebih eksplisit (fully FD hanya pada primary key).
  - :: menghindari update anomali yang masih mungkin terjadi pada 2NF.

### **Note :**

**Jika suatu relasi memenuhi 2NF dan hanya memiliki tepat satu atribut yang bukan kunci utama maka relasi tsb memenuhi 3NF**

## Third Normal Form (3NF)

### Contoh :

Diketahui tabel  $R=(\underline{A},\underline{B},C,D,E)$  ;  $A,B$  kunci utama (primary key) dengan FD :  $A,B \rightarrow C,D,E$  dan  $C \rightarrow D,E$  maka  $R$  bukan 3NF sebab : Atribut  $D$  dan  $E$  (bukan kunci utama) bergantung secara fungsional pada  $C$  (yang juga bukan kunci utama).

### Melalui FD :

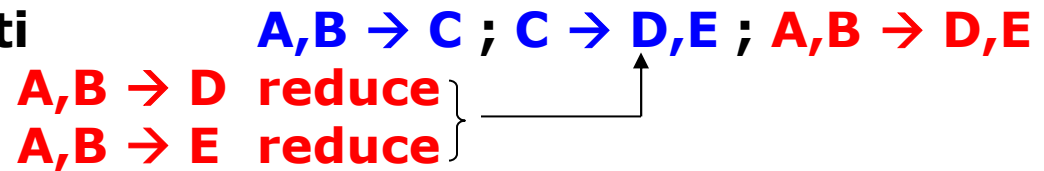
- Diketahui  $A,B \rightarrow C,D,E$ .  
Karena sifat refleksif maka  $A,B \rightarrow A,B$ . Sehingga  $A,B \rightarrow A,B,C,D,E$   
( $A,B$ ) : Superkey.
- Diketahui  $C \rightarrow D,E$ .  
Karena sifat refleksif maka  $C \rightarrow C$ . Sehingga  $C \rightarrow C,D,E$ .  
Karena  $C \not\rightarrow A,B,C,D,E$  maka  $C$  bukan superkey.
- Tidak memenuhi definisi 3NF. Jadi  $R$  bukan 3NF.

Agar  $R$  memenuhi 3NF maka didekomposisi menjadi :  
 $R1=(\underline{A},\underline{B},C)$  dan  $R2=(\underline{C},D,E)$  sehingga  $R1$  dan  $R2$  memenuhi 3NF.

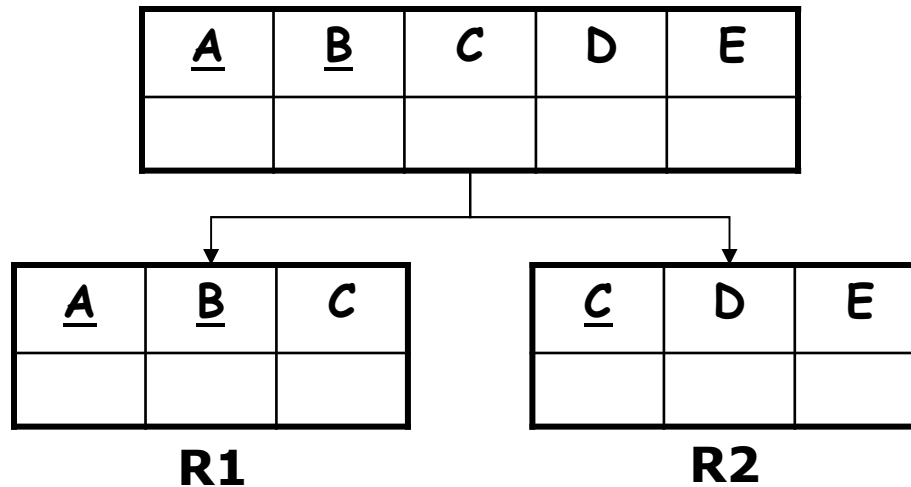
## Third Normal Form (3NF)

FD :  $A, B \rightarrow C, D, E$  berarti

$A, B \rightarrow C$  ;  $C \rightarrow D, E$  ;  $A, B \rightarrow D, E$   
 $A, B \rightarrow D$  reduce  
 $A, B \rightarrow E$  reduce



Dekomposisinya :  $R1 = (\underline{A}, \underline{B}, C)$  ; FD :  $(A, B) \rightarrow C$   
 $R2 = (\underline{C}, D, E)$  ; FD :  $C \rightarrow D, E$





## Third Normal Form (3NF)

**Misal diketahui struktur informasi dari suatu dokumen supplier :**

| S  | Status | City   | PQ |     |
|----|--------|--------|----|-----|
|    |        |        | P  | Qty |
| S1 | 20     | LONDON | P1 | 300 |
|    |        |        | P2 | 200 |
|    |        |        | P3 | 400 |
|    |        |        | P4 | 200 |
|    |        |        | P5 | 100 |
|    |        |        | P6 | 100 |
| S2 | 10     | PARIS  | P1 | 300 |
|    |        |        | P2 | 400 |
| S3 | 10     | PARIS  | P2 | 200 |
| S4 | 20     | LONDON | P2 | 200 |
|    |        |        | P4 | 399 |
|    |        |        | P5 | 400 |

**Akan dibentuk suatu tabel dengan skema**  
**TPS=(S,Status,City,P,Qty)**

**dengan (S,P) = primary key**

**dan berlaku FD :**

**S→Status**

**S→City**

**City→Status**

**Lakukan normalisasi dari 1NF hingga 3NF.**

## Third Normal Form (3NF)

### TPS

| <u>S</u> | Status | City   | <u>P</u> | Qty |
|----------|--------|--------|----------|-----|
| S1       | 20     | LONDON | P1       | 300 |
| S1       | 20     | LONDON | P2       | 200 |
| S1       | 20     | LONDON | P3       | 400 |
| S1       | 20     | LONDON | P4       | 200 |
| S1       | 20     | LONDON | P5       | 100 |
| S1       | 20     | LONDON | P6       | 100 |
| S2       | 10     | PARIS  | P1       | 300 |
| S2       | 10     | PARIS  | P2       | 400 |
| S3       | 10     | PARIS  | P2       | 200 |
| S4       | 20     | LONDON | P2       | 200 |
| S4       | 20     | LONDON | P4       | 399 |
| S4       | 20     | LONDON | P5       | 400 |

- **1NF**
- **Not 2NF**

**Problem :**

- **Redundansi → inconsistency**  
low speed process
- **Anomaly :**  
S→(Status, City) tapi kita tidak bisa insert data (S5, 30, JAKARTA) tanpa diikuti data P (khususnya) dan Q. Menghapus 1 baris data akan jg merusak keutuhan informasi.
- **Solusi :**  
Dekomposisi menjadi :  
**TPS1 dan TPS2**

## Third Normal Form (3NF)

### TPS1

| <u>S</u> | Status | City   |
|----------|--------|--------|
| S1       | 20     | LONDON |
| S2       | 10     | PARIS  |
| S3       | 10     | PARIS  |
| S4       | 20     | LONDON |

- 1NF
- 2NF
- Not 3NF (trans.)  
 $S \rightarrow City$   
 $City \rightarrow Status$

- Sekarang kita dapat menambah data (S5,30,JAKARTA) dgn aman
- Tapi masih ada anomaly :  
 Karena  $City \rightarrow Status$   
 maka kita tidak bisa entry data City baru sebelum Status punya nilai. Penghapusan 1 baris sebagian data City juga bisa merusak keutuhan informasi S.
- Selain itu, masih ada redundansi pada Status dan City

### TPS2

| <u>S</u> | <u>P</u> | Qty |
|----------|----------|-----|
| S1       | P1       | 300 |
| S1       | P2       | 200 |
| S1       | P3       | 400 |
| S1       | P4       | 200 |
| S1       | P5       | 100 |
| S1       | P6       | 100 |
| S2       | P1       | 300 |
| S2       | P2       | 400 |
| S3       | P2       | 200 |
| S4       | P2       | 200 |
| S4       | P4       | 399 |
| S4       | P5       | 400 |

- 1NF
- 2NF
- 3NF  
 redundansi partial  
 → not potensial  
 → better then  
 previous redundant  
**We may not eliminate all redundant but we make its minimize**

## Third Normal Form (3NF)

