

**Basis Data 1 - TIS3333 || 3 SKS**

T.Informatika@2013

- Normalisasi
- Anomali
- Dependensi
- Bentuk Normal

fppt.com

## Normalisasi

- Proses normalisasi adalah proses menganalisa dan memperbaiki skema relasi menggunakan data atribut untuk memperoleh properti-properti skema relasi yang baik menjadi bentuk normal lebih tinggi sehingga dapat:
  - Mengoptimalisasi redundansi
  - Menghilangkan anomali

fppt.com

## Normalisasi #

- Suatu proses untuk mengurangi redundancy pada suatu tabel/obyek/entitas sehingga mudah dimodifikasi.
- Biasanya melibatkan pembagian tabel menjadi dua atau lebih tabel dan menjelaskan hubungan diantara tabel.
- Tujuannya adalah untuk mengisolasi data sehingga penambahan, pengurangan, dan modifikasi dari atribut dapat dilakukan pada satu tabel dan selanjutnya dikembangkan keseluruhan database melalui relasi yang ada.

fppt.com

## Redundancy

- Pengulangan/duplikasi data yang tidak perlu.
- Redundansi tidak dapat dihilangkan sama sekali karena berguna untuk *integritas referensial* yang menghubungkan satu field pada suatu tabel dengan field lain pada tabel yang berbeda.

fppt.com

## Contoh Redundansi

Tabel KRS

NIM	Kode_MK	Nama_MK	Nilai
2013610001	TIS4333	Perancangan Sistem	A
2013610002	TIS4333	Perancangan Sistem	B
2013610003	TIS4333	Perancangan Sistem	B
2013610004	TIS4333	Perancangan Sistem	A
2013610001	TIS3333	Basis Data	B
2013610002	TIS3333	Basis Data	B
2013610003	TIS3333	Basis Data	A
2013610001	TIS1313	Pemrograman	B

- Tabel diatas memiliki redundansi.
- Kolom Kode\_MK dan Nama\_MK memiliki data yang sama (berulang-ulang).

fppt.com

## Anomali

- Anomali pada dasarnya adalah ketidak-konsistenan (inkonsistensi) data.
- Anomali bisa terlihat pada saat melakukan perubahan, penghapusan dan penambahan data.

fppt.com

## Contoh Anomali

Tabel KRS

NIM	Kode_MK	Nama_MK	Nilai
2012610010	TIS3313	Sistem Operasi	A
2012610011	TIS3313	Sistem Operasi	B
2012610014	TIS3313	Sistem Operasi	B
2012610016	TIS3333	Sistem Basis Data	A
2012610020	TIS3333	Basis Data 1	B
2012610021	TIS3333	Basis Data 1	B
2012610023	TIS3333	Basis Data 1	A
2012610024	TIS1313	Bahasa Pemrograman	B

- Misalkan pada tabel diatas dilakukan perubahan data pada record ke 5, data nama mata kuliah *basis data* diganti menjadi *perancangan basis data*.

fppt.com

## Contoh Anomali #

Tabel KRS

NIM	Kode_MK	Nama_MK	Nilai
2012610010	TIS3313	Sistem Operasi	A
2012610011	TIS3313	Sistem Operasi	B
2012610014	TIS3313	Sistem Operasi	B
2012610016	TIS3333	Sistem Basis Data	A
2012610020	TIS3333	Basis Data 1	B
2012610021	TIS3333	Basis Data 1	B
2012610023	TIS3333	Basis Data 1	A
2012610024	TIS1313	Bahasa Pemrograman	B

- Pada tabel diatas akan terjadi ketidak-konsistenan pada record 5, 6 dan 7. yaitu record *Kode\_MK = BD-5452* berelasi dengan dua *Nama\_MK* yang berbeda (Perancangan basis data dengan Basis data).

fppt.com

## ⇒ Solusi Normalisasi

- Melakukan dekomposisi dengan menentukan ketergantungan fungsional (*Functional dependency*).
- Dependensi : suatu hubungan dimana perubahan pada satu tabel akan mempengaruhi tabel yang lain, dimana salah satu tabel bergantung pada tabel yang lain.

- Contoh ketergantungan fungsional :  
Ketergantungan fungsional pada tabel KRS

Kode\_MK                      →            Nama\_MK  
Nim, Kode\_MK                →            Nilai

fppt.com

## Dekomposisi

- Berdasarkan ketergantungan fungsional tersebut, maka tabel KRS dapat didekomposisi menjadi 2 tabel

Kode_MK	Nama_MK
PS-4533	Perancangan Sistem
BD-5630	Sistem Basis Data
BD-5452	Perancangan Basis Data
PG-3465	Pemrograman

NIM	Kode_MK	Nilai
200543423	PS-4533	A
200512345	PS-4533	B
200534632	PS-4533	B
200534632	BD-5630	A
200634123	BD-5452	B
200612367	BD-5452	B
200691245	BD-5452	A
200691245	PG-3465	B

fppt.com

- Dekomposisi tabel dapat mengurangi redundansi yang ada dan menghilangkan anomali
- Perancangan melalui proses normalisasi mempunyai keuntungan-keuntungan sebagai berikut :
  - Meminimalkan ukuran penyimpanan yang diperlukan untuk penyimpanan data.
  - Meminimalkan resiko inkonsistensi data pada basis data.
  - Meminimalkan kemungkinan anomaly pembaruan.
  - Memaksimalkan stabilitas struktur data.

fppt.com

- Tujuan proses normalisasi adalah mengkonversi relasi menjadi bentuk normal lebih tinggi.
- Terdapat beragam tingkat bentuk normal, yaitu:
  1. Bentuk normal pertama (1NF)
  2. Bentuk normal kedua (2NF)
  3. Bentuk normal ketiga (3NF)
  4. Bentuk normal Boyce-Codd (BCNF)
  5. Bentuk normal keempat (4NF)
  6. Bentuk normal kelima (5NF)

fppt.com

- Bentuk normal pertama (1NF) untuk menghilangkan atribut bernilai jamak.
- Bentuk normal kedua (2NF) untuk menghilangkan kebergantungan parsial.
- Bentuk normal ketiga (3NF) untuk menghilangkan kebergantungan transitif.
- Bentuk normal Boyce-Codd (BCNF) untuk menghilangkan anomaly tersisa disebabkan kebergantungan fungsional.
- Bentuk normal keempat (4NF) untuk menghilangkan kebergantungan nilai jamak.
- Bentuk normal kelima (5NF) untuk menghilangkan anomaly tersisa.

fppt.com

- Tiga bentuk normal pertama berkaitan dengan kebergantungan fungsional. Sementara itu bentuk keempat dan kelima berkaitan dengan redundansi yang disebabkan kebergantungan banyak nilai (multi-valued dependencies)

fppt.com

## 1NF

- Syarat 1NF adalah :
  - setiap atribut harus bersifat atomik, artinya, setiap atribut merupakan unsur terkecil dari identitas entitas (tidak perlu dipecah-pecah lagi)
  - semua nilai atributnya adalah sederhana, bukan atribut komposit (composite) dan bukan atribut bernilai banyak (multi-valued).
  - Menghilangkan *repeating group data* (kelompok data yang berulang).

fppt.com

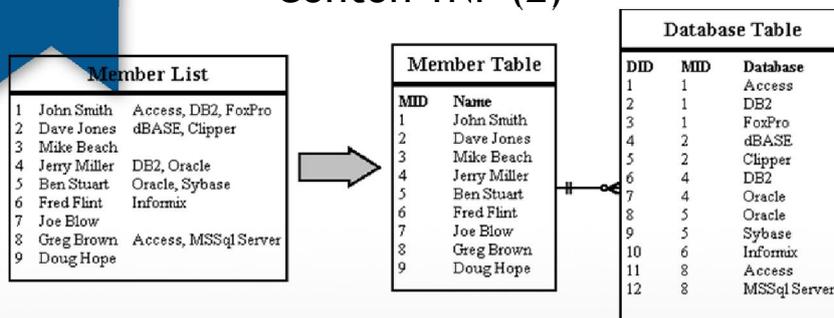
## Contoh 1NF

- Coba perhatikan tabel disamping ini. Apakah tabel Member List ini sudah termasuk dalam 1NF ?

Member List		
1	John Smith	Access, DB2, FoxPro
2	Dave Jones	dBASE, Clipper
3	Mike Beach	
4	Jerry Miller	DB2, Oracle
5	Ben Stuart	Oracle, Sybase
6	Fred Flint	Informix
7	Joe Blow	
8	Greg Brown	Access, MSSql Server
9	Doug Hope	

fppt.com

## Contoh 1NF (2)



- Tabel *member list* memiliki tipe atribut multi-valued
- Sehingga tabel *member list* harus didekomposisikan menjadi 2 tabel.
- Antara *tabel member* dan *database* direlasikan menggunakan atribut *MID* yang merupakan primary-key pada *tabel member* dan foreign-key pada *tabel database*

fppt.com

## 2NF

- Syarat 2NF :
  - telah memenuhi 1NF,
  - setiap atribut *non key* (*bukan primary atau foreign key*) harus tergantung secara fungsional dengan atribut *key*-nya.
- Relasi pada bentuk normal kedua harus tidak menyimpan fakta-fakta mengenai bagian kunci relasi. Bentuk normal kedua menghilangkan ketergantungan parsial dan masih memiliki anomali-anomali yang secara praktis tidak dapat diterima.

fppt.com

- Untuk memenuhi syarat 2NF, maka atribut non key harus memiliki ketergantungan fungsional dengan atribut key.
- Tabel harus dipisah menjadi tiga tabel yang memiliki atribut key dan ketergantungan fungsional dengan atribut non key.

fppt.com

## Contoh 2NF

- Tabel Berikut . Apakah tabel ini sudah termasuk dalam 2NF ?

Member Table		Database Table		
MID	Name	DID	MID	Database
1	John Smith	1	1	Access
2	Dave Jones	2	1	DB2
3	Mike Beach	3	1	FoxPro
4	Jerry Miller	4	2	dBASE
5	Ben Stuart	5	2	Clipper
6	Fred Flint	6	4	DB2
7	Joe Elow	7	4	Oracle
8	Greg Brown	8	5	Oracle
9	Doug Hope	9	5	Sybase
		10	6	Informix
		11	8	Access
		12	8	MSSql Server

fppt.com

## Contoh 2NF

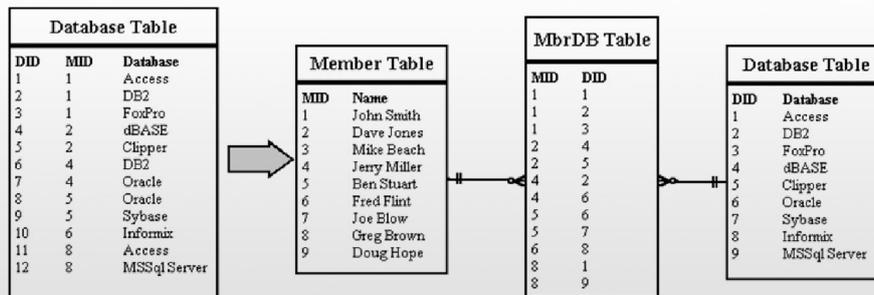
- Didalam tabel database terdapat satu atribut non-key yaitu *atribut database* yang memiliki ketergantungan terhadap *atribut DID* dan *MID*.
- Untuk memenuhi syarat 2NF, maka *atribut database* di set up sehingga tergantung dengan primary key yang sesuai yaitu *atribut DID*.

Database Table			Member Table		Database Table	
DID	MID	Database	MID	Name	DID	Database
1	1	Access	1	John Smith	1	Access
2	1	DB2	2	Dave Jones	2	DB2
3	1	FoxPro	3	Mike Beach	3	FoxPro
4	2	dBASE	4	Jerry Miller	4	dBASE
5	2	Clipper	5	Ben Stuart	5	Clipper
6	4	DB2	6	Fred Flint	6	Oracle
7	4	Oracle	7	Joe Elow	7	Sybase
8	5	Oracle	8	Greg Brown	8	Informix
9	5	Sybase	9	Doug Hope	9	MSSql Server
10	6	Informix				
11	8	Access				
12	8	MSSql Server				

fppt.com

## Contoh 2NF

- Untuk menghubungkan tabel database dengan tabel member, maka dibuat suatu tabel baru yang memiliki atribut dari kedua tabel utama.



fppt.com

## 3NF

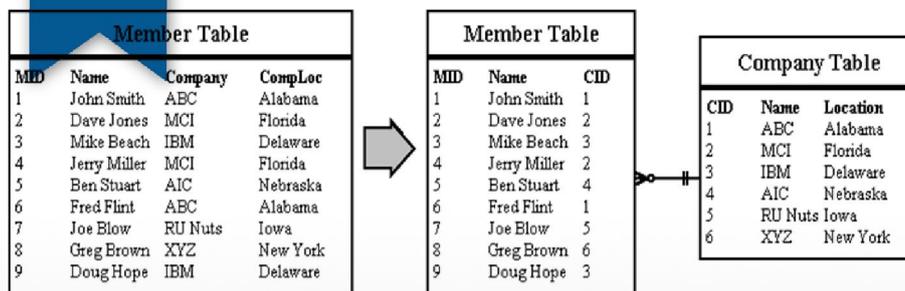
- Syarat 3NF :
  - telah memenuhi 2NF
  - setiap atribut *non key* tidak boleh tergantung dengan atribut *non key* lainnya (tidak boleh terjadi ketergantungan transitif).

fppt.com

- Pada tabel Member terdapat satu primary key yaitu MID.
- Atribut non key lainnya yaitu *Name*, *Company* terhubung dengan primary key MID.

MID	Name	Company	CompLoc
1	John Smith	ABC	Alabama
2	Dave Jones	MCI	Florida
3	Mike Beach	IBM	Delaware
4	Jerry Miller	MCI	Florida
5	Ben Stuart	AIC	Nebraska
6	Fred Flint	ABC	Alabama
7	Joe Blow	RU Nuts	Iowa
8	Greg Brown	XYZ	New York
9	Doug Hope	IBM	Delaware

fppt.com



- Atribut CompLoc sebagai atribut non-key tidak tergantung secara fungsional dengan primary key MID.
- Atribut CompLoc (non-key) memiliki ketergantungan dengan atribut Company (non-key), ketergantungan antara non-key dengan non-key disebut ketergantungan transitif.
- Untuk mengatasi masalah ini, dibuat tabel baru yaitu tabel Company.

fppt.com

## Bentuk Normal Boyce-Codd (BCNF)

- BCNF memiliki ketentuan yaitu :
  - masing-masing atribut utama bergantung fungsional penuh pada masing-masing kunci dimana kunci tersebut bukan bagiannya.
  - Relasi adalah BCNF (optimal) jika setiap determinan atribut-atribut relasi adalah kunci relasi.
  - Relasi adalah BCNF (optimal) jika kapanpun fakta-fakta disimpan mengenai beberapa atribut, maka atribut-atribut ini merupakan satu kunci relasi.
  - BCNF dapat memiliki lebih dari satu kunci.
  - Properti penting BCNF adalah relasi tidak memiliki informasi yang redundan.

fppt.com

- Ada dua kasus yang membuat 3NF harus dijadikan BCNF :
  - Dalam satu entity terdapat lebih dari satu atribut key, dan masing-masing dari key menentukan atributnya masing-masing.
  - Ketergantungan terjadi dari atribut non key ke atribut key, Atribut non key menentukan bagian dari atribut key.

fppt.com