

NORMALISASI

Meri Azmi, ST

Normalisasi Data

- ▶ Dalam merancang basis data, dapat dilakukan dengan:
 - ▶ Menerapkan Normalisasi
 - ▶ Membuat model ER (Entity Relationship)

created by : meri@polinpdg.ac.id

TODAY !

- ▶ Kebergantungan Fungsional
- ▶ Dekomposisi
- ▶ Proses Normalisasi

created by : meri@polinpdg.ac.id

Normalisasi

- ▶ Proses normalisasi dalam basis data adalah proses untuk memperoleh properti skema Tabel yang bagus dan efisien serta terhindar dari anomali.
- ▶ Keuntungan normalisasi :
 - ▶ Membuat sekecil mungkin terjadinya data rangkap.
 - ▶ Menghindarkan adanya data yang tidak konsisten bila dilakukan penghapusan / penambahan data.
 - ▶ Menjamin bahwa key tabel secara tunggal sebagai determinan semua atribut.

created by : meri@polinpdg.ac.id

Anomali

- ▶ Anomali adalah proses pada basis data yang memberikan efek samping yang tidak diharapkan, misalnya menyebabkan ketidak konsistenan data atau membuat sesuatu data menjadi hilang ketika data lain dihapus.

created by : meri@polinpdg.ac.id

Anomali

- ▶ Anomali Peremajaan (Update / Edit)
terjadi apabila terjadi pengubahan terhadap sejumlah data yang mubazir, tetapi tidak seluruhnya di ubah
- ▶ Anomali Penyisipan (Insert)
terjadi jika pada saat penambahan hendak dilakukan ternyata ada elemen data yang masih kosong dan elemen tersebut justru menjadi kunci
- ▶ Anomali Penghapusan (Delete)
terjadi sekiranya sesuatu baris (tuple) yang tak terpakai dihapus dan sebagai akibatnya terdapat data lain yang akan hilang

created by : meri@polinpdg.ac.id

Hal Penting dalam Normalisasi

- ▶ Kebergantungan Fungsional
- ▶ Dekomposisi

created by : meri@polinpdg.ac.id

Kebergantungan Fungsional

- ▶ Menunjukkan satu nilai unik pada suatu atribut ditentukan oleh suatu nilai atribut lain.
- ▶ Contoh :
Seorang pegawai dengan **IDPegawai** mempunyai satu **tglLahir** dapat dipresentasikan dgn KF berikut :
IDPegawai → tglLahir

Keterangan :

- ▶ Sisi kiri adalah determinan (penentu)
- ▶ Nilai di determinan dapat menentukan hanya satu nilai sisi kanan.
- ▶ Nilai **IDPegawai** menentukan satu nilai **tglLahir**.

Kebergantungan Fungsional

- ▶ Ada tiga jenis KF:
 - ▶ Saling Bergantung.
 - ▶ Kebergantungan lebih dari satu atribut.
 - ▶ Kebergantungan fungsional penuh.

created by : meri@polinpdg.ac.id

Kebergantungan Fungsional (Saling Bergantung)

- ▶ Contoh :

Jika satu proyek mempunyai satu manajer dan masing-masing manajer hanya mengelola satu proyek, maka dapat dipresentasikan sebagai berikut :

manajer → IDProyek
IDProyek → manajer

Atau dapat disingkat

manajer ↔ IDProyek

created by : meri@polinpdg.ac.id

Kebergantungan Fungsional (Bergantung lebih dari 1 atribut)

► Contoh :

waktuketerlibatan pegawai disuatu proyek, ditentukan oleh **IDPegawai** dan **IDProyek**. Dipresentasikan sebagai berikut:

IDPegawai, IDProyek → waktuketerlibatan

created by : meri@polinpdg.ac.id

Kebergantungan Fungsional Penuh

- Adalah KF dimana tidak ada atribut-atribut tak perlu yang berada disisi determinan (sisi kiri).

Contoh :

- **IDPegawai → tglLahir**
- **IDPegawai, nama → tglLahir**
- Pada contoh 2, **nama** tidak diperlukan untuk memperoleh **tglLahir**. **IDPegawai** telah mencukupi untuk memperoleh nilai **tglLahir**.
- Sehingga contoh 1 adalah KF penuh dan contoh 2 bukan KF penuh.

created by : meri@polinpdg.ac.id

Dekomposisi

- ▶ Yaitu melakukan pemilahan / pemisahan tabel menjadi beberapa tabel dengan mempertimbangkan KF yang telah didapat.
- ▶ Dekomposisi dilakukan agar setiap tabel yang digunakan hanya memiliki satu KF saja.
- ▶ Dekomposisi yang benar (Lossless Decomposition) terjadi jika tabel hasil dekomposisi digabungkan kembali dapat menghasilkan tabel awal sebelum didekomposisi.

created by : meri@polinpdg.ac.id

Dekomposisi

- ▶ Contoh :
Diberikan sebuah tabel ABC
- ▶ Tabel ABC didekomposisi menjadi tabel AB dan BC.

A	B	C
A1	100	C1
A2	200	C2
A3	300	C3
A4	200	C2

A	B
A1	100
A2	200
A3	300
A4	200

B	C
100	C1
200	C2
300	C3

created by : meri@polinpdg.ac.id

Proses Normalisasi

- ▶ Tabel (tabel) secara berurutan diproses menjadi bentuk normal lebih tinggi yang memiliki redundansi lebih rendah.

created by : meri@polinpdg.ac.id

Proses Normalisasi

- ▶ Terdapat beragam tingkat bentuk normal :
 - ▶ Bentuk normal pertama (1NF).
 - ▶ Bentuk normal kedua (2NF).
 - ▶ Bentuk normal ketiga (3NF).
 - ▶ Bentuk normal Boyce-Codd (BCNF).
 - ▶ Bentuk normal keempat (4NF).
 - ▶ Bentuk normal kelima (5NF).

created by : meri@polinpdg.ac.id

Bentuk Normal Pertama (1NF)

- ▶ Dicapai bila tiap nilai atribut adalah tunggal (bukan atribut turunan).
- ▶ Kondisi ini dapat diperoleh dengan melakukan eliminasi terjadinya data rangkap.

created by : meri@polinpdg.ac.id

Bentuk Normal Pertama (1NF)

- ▶ Contoh :
Tabel **Pesanan**.

IdPesanan	TglPesanan	isiPesanan	
T20	6 Jan 2016	IdItem	Kuantitas
		PC6	24
		BW3	83
		TY6	37
T33	5 Feb 2016	IdItem	Kuantitas
		PC6	810
		BW3	87
		TY6	810
		HJ4	5

created by : meri@polinpdg.ac.id

Tabel pesanan diatas tidak 1NF karena nilai **isiPesanan** tidak tunggal, sehingga harus dinormalisasi.

Bentuk Normal Pertama (1NF)

- ▶ Setelah melalui proses normalisasi pertama, tabel **Pesanan** menjadi :

IdPesanan	TglPesanan	IdItem	Kuantitas
T20	6 Jan 2016	PC6	24
T20	6 Jan 2016	BW3	83
T20	6 Jan 2016	TY6	37
T33	5 Feb 2016	PC5	810
T33	5 Feb 2016	BW387	87
T33	5 Feb 2016	TY6	810
T33	5 Feb 2016	HJ4	5

created by : meri@polinpdg.ac.id

Bentuk Normal Kedua (2NF)

- ▶ Semua atribut bukan utama harus bergantung fungsional penuh pada primary key.
- ▶ Tabel harus telah berbentuk normal pertama.

created by : meri@polinpdg.ac.id

Bentuk Normal Kedua (2NF)

IdPekerja	IdProyek	Dana Proyek	Waktu Keterlibatan
P001	PJ001	20	20
P003	PJ001	20	16
P002	PJ002	17	35
P002	PJ003	84	42
P003	PJ002	17	17
P004	PJ001	20	83
P004	PJ003	84	41
-	PJ004	100	-

created by : meri@polinpdg.ac.id

- ▶ Contoh :
Tabel **Data_Projek**
- ▶ Skema Tabel :
Data_Projek =
{ {IdPekerja, IdProyek, DanaProyek, WaktuKeterlibatan},
{IdPekerja, IdProyek → WaktuKeterlibatan,
IdProyek → DanaProyek} }
- ▶ Primary Key : **IDPekerja, IDProyek**
- ▶ Atribut2 bukan utama : **DanaProyek** dan **WaktuKeterlibatan**
- ▶ **WaktuKeterlibatan** tergantung penuh pada Primary Key.
- ▶ **DanaProyek** tidak bergantung penuh terhadap Primary Key.
- ▶ Tabel **Data_Projek** belum memenuhi 2NF, sehingga perlu dinormalisasi dengan dekomposisi.

Bentuk Normal Kedua (2NF)

IdProyek	DanaProyek
PJ001	20
PJ002	17
PJ003	84
PJ004	100

IdPekerja	IdProyek	Waktu Keterlibatan
P001	PJ001	20
P003	PJ001	16
P002	PJ002	35
P002	PJ003	42
P003	PJ002	17
P004	PJ001	83
P004	PJ003	41

created by : meri@polinpdg.ac.id

- ▶ Tabel **Data_Projek** didekomposisi menjadi Tabel **Proyek** dan **Pekerjaan**.
- ▶ Skema Tabel :
 ▶ Tabel **Proyek** =
 { {IdProyek, DanaProyek},
 {IdProyek → DanaProyek} }
 ▶ Tabel **Pekerjaan** =
 { {IdPekerja, IdProyek, WaktuKeterlibatan},
 {IdPekerja, IdProyek → WaktuKeterlibatan} }

Bentuk Normal Ketiga (3NF)

- ▶ Tabel telah berbentuk normal kedua.
- ▶ Tabel tidak boleh memuat kebergantungan fungsional diantara atribut-atribut bukan utama.
- ▶ Bentuk 3NF mengilangkan kebergantungan transitif.

created by : meri@polinpdg.ac.id

Bentuk Normal Ketiga (3NF)

IdProyek	Manajer	TglLahir
PJ001	Jon	24 Jan 1980
PJ002	Viki	23 Mar 1978
PJ003	Jon	24 Jan 1980
PJ004	Marini	6 Juli 1981

- Contoh :
Tabel **Proyek**
- Skema Tabel :
Proyek =
{IdProyek, Manajer, TglLahir},
{IdProyek → Manajer,
IdProyek → TglLahir,
manajer → TglLahir})
- Primary Key : **IDProyek**
- Terdapat ketergantungan antara atribut bukan utama.
- Sehingga Tabel **proyek** bukan 3NF dan perlu didekomposisi.

created by : meri@polinpdg.ac.id

Bentuk Normal Ketiga (3NF)

IdProyek	Manajer
PJ001	Jon
PJ002	Viki
PJ003	Jon
PJ004	Marini

Manajer	TglLahir
Jon	24 Jan 1980
Viki	23 Mar 1978
Marini	6 Juli 1981

created by : meri@polinpdg.ac.id

- Tabel **Proyek** didekomposisi menjadi Tabel **Proyek** dan **Manajer**.

- Skema Tabel :

- Tabel **Proyek** =
{ {IdProyek, Manajer},
{IdProyek → Manajer} }
- Tabel **Manajer** =
{ {Manajer, TglLahir},
{Manajer → TglLahir} }

Tugas Kelompok (Maks 3 orang / kelompok)

- ▶ Perhatikan **tabel universal** (Hal:52, Buku Basis Data, Fathansyah).
- ▶ Lakukan proses normalisasi pada tabel tersebut.
- ▶ Buatlah laporan tentang proses normalisasi tersebut.
- ▶ Presentasikan.

created by : meri@polinpdg.ac.id

Boy Cod Normal Form (BCNF)

- ▶ Masing-masing atribut bukan utama bergantung fungsional penuh pada masing-masing kunci, dimana kunci tersebut bukan bagiannya.

created by : meri@polinpdg.ac.id

Bentuk Normal Tinggi

- ▶ Tabel bisa menimbulkan kebergantungan banyak nilai (multivalued dependency).
- ▶ Hal ini bisa diatasi dengan bentuk normal tinggi, yaitu :
 - ▶ Bentuk normal keempat (4NF).
 - ▶ Bentuk normal kelima (5NF).

created by : meri@polinpdg.ac.id

Kebergantungan Banyak Nilai (Multivalued Dependency)

► Diberikan sebuah Tabel :

Mata Kuliah	Instruktur	Bukuteks
Manajemen	Wisnu Panji Ratih Nina	Drucker Peters
Keuangan	Asri	Jones Chang

created by : meri@polinpdg.ac.id

Kebergantungan Banyak Nilai

► Maka bentuk BCNF dari Tabel diatas adalah :

MataKuliah	Instruktur	BukuTeks
Manajemen	Wisnu	Drucker
Manajemen	Wisnu	Peters
Manajemen	Panji	Drucker
Manajemen	Panji	Peters
Manajemen	Ratih	Drucker
Manajemen	Ratih	Peters
Manajemen	Nina	Drucker
Manajemen	Nina	Peters
Keuangan	Asri	Jones
Keuangan	Asri	Chang

created by : meri@polinpdg.ac.id

Bentuk Normal Keempat (4NF)

- ▶ Tabel telah dalam bentuk BCNF.
- ▶ Tidak berisi kebergantungan banyak nilai.
- ▶ Tabel diatas belum memenuhi 4NF, oleh sebab itu harus didekomposisi.

created by : meri@polinpdg.ac.id

Bentuk Normal Keempat (4NF)

MataKuliah	Instruktur	MataKuliah	BukuTeks
Manajemen	Wisnu	Manajemen	Druckers
Manajemen	Panji	Manajemen	Peters
Manajemen	Ratih	Keuangan	Jones
Manajemen	Nina	Keuangan	Chang
Keuangan	Asri		

created by : meri@polinpdg.ac.id

Bentuk Normal kelima (5NF) / Project Point Normal Form (PJNF)

- ▶ Berkaitan dengan ketergantungan pada gabungan beberapa atribut.
- ▶ Kasus ini dalam praktek jarang ditemukan, sebab secara nyata akan disusun suatu tabel dengan atribut-atribut yang ditentukan dengan satu identitas.

created by : meri@polinpdg.ac.id