

## PERTEMUAN 14

### NORMALISASI

#### A. TUJUAN PEMBELAJARAN

Setelah mempelajari dengan baik materi ini, diharapkan semua mahasiswa mampu memahami materi yang diberikan:

1. Untuk menghilangkan kerangkapan data atau double data
2. Untuk mengurangi kompleksitas data
3. Untuk mempermudah pemodifikasian data yang ada

#### B. URAIAN MATERI

##### 1. Dekomposisi 1NF

Bentuk 1NF mempunyai ciri yaitu setiap data dibentuk dalam flat *file* (*file* datar/rata) informasi dibingkai dalam catatan oleh satu catatan dan nilai bidang adalah “atomic value” tidak ada set atribut yang berlebihan atau atribut bernilai ganda (multivalue). Tiap *field* hanya memiliki satu makna, bukan merupakan kumpulan kata yang mempunyai arti mendua, hanya satu arti saja dan juga bukanlah pecahan kata kata sehingga artinya lain.

Atom adalah zat terkecil yang masih memiliki sifat induknya bila dipecah lagi maka ia tidak akan memiliki sifat induknya.

Misalnya :

KELAS (kode\_kelas,nama\_kelas,instruktur)

Merupakan bentuk 1NF karena tidak ada yang berganda dan tiap atribut satu pengertian tunggal.

Contoh data

**Tabel 0.1** Kelas (1NF)

kode_kelas	nama_kelas	instruktur
1234	fisika dasar	suroso
1543	matematika	paulus
1775	teknik program	bagus

SISWA (NO\_siswa, Nama, Wali\_studi, kelas1, kelas2, kelas3)

Siswa yang punya tiga nomor siswa, nama dan wali studi mengikuti 3 mata pelajaran/kelas. Disini ada perulangan kelas 3 kali ini bukan bentuk 1NF.

**Tabel 0.2** Unnormalized

No_siswa	nama	wali_studi	kelas1	kelas2	kelas3
22890100	Tanzania	Zaman	1234	1543	
22890101	Nia	Rizki	1234	1775	1543

Contoh data:

Bentuk normal ke satu dari bentuk diatas

**Tabel 0.3** Siswa (1NF)

No_siswa	nama	wali_studi	kode_kelas
22890100	Tanzania	Zaman	1234
22890100	Tanzania	Zaman	1543
22890101	Nia	Rizki	1234
22890101	Nia	Rizki	1775
22890101	Nia	Rizki	1543

Contoh lain:

Berikut ini adalah contoh data-data yang belum ternormalisasi.

**Tabel 0.4** Relasi Pegawai

NIP	nama	jabatan	keahlian	Lama (tahun)
1993	Daffa	programer senior	java mobile	5
			android	4
2003	Revan	programer yunior	php laravel	3
			android	2
2008	Lintang	programer senior	php laravel	1
			arduino	2
			java mobile	1

Keahlian menyatakan atribut yang berulang (missal, Lintang punya tiga keahlian, dan Daffa/Revan punya 2 keahlian).

Berikut ini adalah contoh data pada relasi pegawai yang telah memenuhi bentuk normal pertama.

**Tabel 0.5** Relasi Pegawai (1NF)

NIP	nama	jabatan	keahlian	Lama (tahun)
1993	Daffa	programer senior	java mobile	5
1993	Daffa	programer senior	android	4
2003	Revan	programer yunior	php laravel	3

2003	Revan	programer yunior	android	2
2008	Lintang	programer senior	php laravel	1
2009	Lintang	programer senior	arduino	2
2010	Lintang	programer senior	java mobile	1

Berikut ini adalah contoh data pada relasi mahasiswa yang belum memenuhi bentuk normal

**Tabel 0.6** Relasi Mahasiswa

NIM	Nama	Dosen wali	Kode_mk1	Kode_mk2	Kode_mk3
2001	jafar sadiq	didik setiyadi	1234	1435	
1004	rully p	rita wahyuni	1234	1435	1245
1006	nadya safitri	fata nidaul	1234	1435	1245
2008	syahbaniar	endang	1234	1435	

Relasi mahasiswa yang mempunyai NIM, nama dan dosen wali megikuti 3 mata kuliah, tabel tersebut belum memenuhi normal pertama karena ada perulangan kode\_mk 3 kali padahal hal tersebut bisa dijadikan 1 atribut saja. Jadi bentuk normal pertama dari data diatas adalah:

**Tabel 0.7** Relasi Mahasiswa Memenuhi 1NF

NIM	Nama	Dosen wali	Kode_matakuliah
2001	jafar sadiq	didik setiyadi	1234
2001	jafar sadiq	didik setiyadi	1435
1004	rully p	rita wahyuni	1234
1004	rully p	rita wahyuni	1435
1004	rully p	rita wahyuni	1245
1006	nadya safitri	fata nidaul	1234
1006	nadya safitri	fata nidaul	1435
1006	nadya safitri	fata nidaul	1245
2008	syahbaniar	endang	1234
2008	syahbaniar	endang	1435

Berikut ini adalah contoh data pada relasi matakuliah yang telah memenuhi bentuk normal pertama.

**Tabel 0.8** Relasi Matakuliah Memenuhi 1NF

kode_mk	matakuliah	sks	dosen pengampu
1234	sistem basis data	2	didik setiyadi
1435	algoritma dan pemrograman	4	rita wahyuni
1245	interaksi manusia dan komputer	2	fata nidaul

Relasi matakuliah tersebut merupakan bentuk 1NF karena tidak ada atribut yang bernilai ganda dan tiap atribut satu pengertian yang bernilai tunggal.

Bentuk 1NF, jika pada entitas tidak ada atribut yang dapat memiliki lebih dari satu penggambaran elemen tunggal atau dapat juga dikatakan jika dan hanya semua area mengandung harga atomik. Tiap atribut pada entitas dapat mempunyai nilai ganda yang digambarkan secara aktual pada entitas yang terpisah, mungkin sebuah entitas dan relasinya.

Suatu hubungan dianggap sebagai struktur tipikal pertama jika dan hanya jika setiap kualitas adalah satu-mungkin untuk setiap kolom. Setiap bidang hanya memiliki satu kepentingan, itu tidak lain adalah bermacam-macam kata yang hanya memiliki satu arti dan kata-kata diulang atau memiliki sifat yang berbeda.

## 2. Dekomposisi 2NF

Bentuk normal ke dua dapat dibentuk jika struktur *level* tipikal seperti yang sekarang ada atau dibingkai dan jika nilai semua kunci penting tidak ada di kunci penting (*primary key*) bergantung pada kunci primer, atau sangat baik dapat dikatakan jika dan hanya jika ada 1NF dan tiap atribut non kunci bergantung penuh pada kunci primer. Tiap atribut bukan kunci yang bergantung hanya pada bagian kunci primer harus dipindahkan ke tiap entitas, dimana bagian kunci merupakan kunci penuh yang sebenarnya. Disini mungkin dibutuhkan pembuatan entitas baru dan relasi pada model.

Struktur normal tahap kedua (2NF) merupakan kelanjutan dari struktur tipikal tahap utama (1NF) dalam pemeriksaan informasi.

Perlu diingat Kembali bahwa 2NF membutuhkan semua penempatan entitas yang tersedia dari bentuk normal tahap pertama (1NF). Perlu diingat juga bahwa tahap 2NF dilihat pada siapa yang mempunyai nilai atribut dapat ditentukan hanya oleh bagian pada kunci primer dan tidak digunakan pada rangkaian semua kunci. Begitu pula dengan entitas hanya mempunyai satu atribut kunci primer yang tersedia dalam bentuk tahap ke dua (2NF).

Definisi bentuk normal ke dua (2NF) adalah:

- a. Terpenuhi bentuk struktur 1NF
- b. Atribut non kunci bergantung secara fungsi pada kunci esensial.

Sehingga untuk membentuk normal kedua setiap tabel/rekaman harus diselesaikan kunci-kunci atributnya. Kunci atribut haruslah unik dan dapat mewakili atribut lainnya yang menjadi anggotanya.

Bentuk normal kedua mempunyai syarat memiliki kondisi bahwa struktur informasi telah memenuhi normal ke satu. Karakteristik non-kunci secara praktis bergantung pada kunci esensial. Sehingga untuk membentuk normal kedua penentuan kunci kunci *field* telah diselesaikan.

Contoh:

Pada contoh relasi siswa pada bentuk normal kesatu, terlihat bahwa kunci utama/ primay key adalah nomor siswa. Nama siswa dan wali studi bergantung pada fungsi No\_siswa, tetapi kode\_kelas bukanlah fungsi dari siswa maka *file* siswa dipecah menjadi relasi yaitu:

Relasi siswa

**Tabel 0.9** Relasi Siswa

No_siswa	nama	wali_studi
22890100	Tanzania	Zaman
22890101	Nia	Rizki

Relasi ambil kelas

**Tabel 0.10** Relasi Kelas

No_siswa	kode_kelas
22890100	1234
22890100	1543
22890101	1234
22890101	1775
22890101	1543

Contoh lain:

Pada tabel Mahasiswa yang memenuhi normal pertama (1NF) terlihat bahwa NIM merupakan primary key (PK).

NIM → Nama, Dosen wali. Artinya adalah bahwa atribut nama dan dosen wali bergantung pada NIM.

Tetapi NIM → kode\_matakuliah. Artinya adalah bahwa atribut Kode\_mk tidak tergantung pada NIM.

Untuk memenuhi normal kedua, maka pada relasi mahasiswa tersebut dipecah menjadi 2 relasi sebagai berikut:

**Tabel 0.11** Relasi Mahasiswa Memenuhi 2NF

NIM (PK)	Nama	Dosen wali
2001	jafar sadiq	didik setiyadi
1004	rully p	rita wahyuni
1006	nadya safitri	fata nidaul
2008	syahbaniar	endang



**Tabel 0.12** Relasi Ambil Kuliah Memenuhi 2NF

NIM (FK/PK)	Kode_matakuliah (FK/PK)
2001	1234
2001	1435
1004	1234
1004	1435
1004	1245
1006	1234
1006	1435
1006	1245
2008	1234
2008	1435

### 3. Dekomposisi 3NF

Bentuk normal tahap 3 (3NF) terjadi jika 2NF telah tersedia dan jika nilai atribut yang bukan kunci primer tidak bergantung pada atribut yang bukan kunci primer lainnya atau dapat pula dikatakan sebagai bila dan hanya bila ada dalam 2NF dan tiap atribut non kunci tergantung non transitif atas kunci primer. Bentuk normal tahap ke tiga (3NF) dikenal pula dengan dengan sebutan boyce-codd normal form (BCNF).

3NF dalam model data mempunyai masalah, tidak memenuhi untuk relasi yang:

- a. Mempunyai banyak kunci kandidat.
- b. Kunci kandidat saling tumpang tindih (mempunyai paling tidak satu atribut bersama).

Tiap atribut yang bukan kunci bergantung pada atribut bukan kunci yang lain harus dipindahkan atau dihapus. Sekali lagi, entitas baru dan relasinya mungkin dapat ditambahkan untuk model data.

Kita dapat menyederhanakan lebih lanjut entitas-entitas dengan ditematkannya kedalam 3NF. Semua kebutuhan entitas dalam 2NF harus diidentifikasi sebelum memulai analisis 3NF. Analisis 3NF dapat dilihat pada 2 (dua) tipe masalah, yaitu pengambilan data dan ketegantungan-ketegantungan transitif. Dalam kasus kasus sebelumnya kesalahan secara fundamental adalah ketegantungan pada atribut yang tidak mempunyai kunci. Langkah awal pada tipe 3NF adalah sangat mudah, yaitu menguji atau memeriksa setiap entitas untuk pengambilan data.

Yang dimaksud pengambilan data di sini adalah dimana salah satu nilai dapat dikalkulasi dari atribut atribut lain atau pengiriman logika secara langsung dari nilai pada atribut lainnya.

Jika anda berpikir tentang pengiriman data, cerita pengiriman data membuat sedikit pengertian. Pertama, tentang pemborosan ruang penyimpanan disk. Kedua, kesulitan pengiriman basis data apakah akan mudah untuk diperbaharui (*update*). Mengapa demikian? karena tiap anda mengubah dasar atribut-atribut, anda harus melakukan Kembali penghitungan dan kemudian akan mengubah hasil pengiriman data.

Untuk menjadi bentuk normal ketiga maka relasi haruslah dalam struktur normal kedua dan semua atribut non-kunci esensial tidak memiliki hubungan transitif. Dengan demikian, setiap atribut non-kunci didasarkan secara eksklusif pada kunci esensial dan kunci esensial secara keseluruhan.

Definisi bentuk normal ke tiga (3NF) adalah:

- a. Terpenuhi struktur 2 NF (normal kedua)
- b. Atribut non-kunci tidak memiliki dependensi transtatif terhadap kunci esensial.

Berikut contoh relasi yang memenuhi bentuk 2NF, tetapi tidak memenuhi bentuk 3NF.

**Tabel 0.13** Relasi yang Memenuhi 2NF

NO_Pesanan	Kode_Barang	Kode_Item	Nama Item	Jumlah
50001	br00001	P1	Pensil	5
50001	br00002	P2	Buku Tulis	7
50001	br00003	P3	Penggaris	2
50001	br00004	P4	Penghapus	5
50002	br00001	P3	Penggaris	3
50002	br00002	P5	Bulpen	2
50002	br00003	P6	Spidol	5
50003	br00001	P1	Pensil	6
50003	br00002	P2	Buku Tulis	9

Atribut No\_Pesanan dan Kode\_Barang merupakan kunci primer baik kode item dan nama item mempunyai dependensi fungsional terhadap kunci primer tersebut.

Pada relasi diatas, setiap kode item sama, maka nilai nama item juga sama, sehingga menunjukkan adanya dependensi dua atribut tersebut, tapi manakah yang menentukan, apakah kode item bergantung pada nama item atau sebaliknya? Jadi nama item memiliki dependensi fungsional terhadap kode item.

Pada relasi ini menunjukkan bahwa nama item tidak memiliki dependensi secara langsung terhadap terhadap kunci primer (NO\_Pesanan dan Kode\_barang). Dengan kata lain Nama item memiliki dependensi transitif terhadap kunci primer.

Sehingga untuk memenuhi bentuk 3NF, maka relasi diatas didekomposisi menjadi dua buah relasi sebagai berikut:

**Tabel 0.14** Relasi Pesanan Barang Memenuhi 3NF

NO_Pesanan	Kode_Barang	Kode_Item	Jumlah
50001	br00001	P1	5
50001	br00002	P2	7
50001	br00003	P3	2
50001	br00004	P4	5
50002	br00001	P3	3
50002	br00002	P5	2
50002	br00003	P6	5
50003	br00001	P1	6
50003	br00002	P2	9

**Tabel 0.15** Relasi Barang Memenuhi 3NF

Kode_Item	Nama Item
P1	Pensil
P2	Buku Tulis
P3	Penggaris
P4	Penghapus
P3	Penggaris
P5	Bulpen
P6	Spidol
P1	Pensil
P2	Buku Tulis

#### 4. Penerapan Bentuk Normalisasi dari 1NF hingga 3NF

Pada proses perancangan *database* dapat dimulai dari dokumen dasar yang dipakai dalam sistem sesuai dengan lingkup sistem yang akan dibuat rancangan databsnya. Berikut ini adalah contoh dokumen dari bon pembelian pada “Honda revanda”:

Honda revanda	no bon : 05103214				
AHAS 06488	tanggal : 25/10/2018				
jatimulya-bekasi					
telp. 021-82432162					
<b>BON PENJUALAN</b>					
No polisi :                    B1234LB, id_warna : w01, warna : biru  Id_merk :                      mr01, merek : supra x, tahun : 2005  id_mekanik : SP, Nama_mekanik : supriadi					
kode part	nama part	kuantum	harga	discount	jumlah
20W501000cc	oli federal 1000cc	1	27000	1000	26000
serv001	service mesin	1	25000	500	24500
total					50500
potongan					500
total bayar					50000
idkasir : ksr01, kasir : Dewi					

Honda revanda	no bon : 05103218
AHAS 06488	tanggal : 26/10/2018
jatimulya-bekasi	
telp. 021-82432162	

BON PENJUALAN					
No polisi : B1235LB, id_warna : w01, warna : biru					
Id_merk : mr11, merek : vario, tahun : 2010					
id_mekanik : WN, Nama_mekanik : wawan					
kode part	nama part	kuantum	harga	discount	jumlah
20W501000cc	oli federal 1000cc	2	27000	1000	52000
serv001	service mesin	1	25000	500	24500
kam001	kampus kopling	1	76000	5000	71000
total					147500
potongan					1500
total bayar					146000

idkasir : ksr02, kasir : aura

**Gambar 0.1** Gambar bon pembelian pada “Honda revanda”

sehubungan dengan dokumen dasar tersebut, tahapan yang harus dilakukan untuk melakukan normalisasi adalah sebagai berikut :

a. Bentuk Unnormalisasi

Langkah pertama dalam melakukan normalisasi data adalah dengan membentuk contoh data diatas dengan membentuk unnormalisasi data, dengan cara mencantumkan atribut data yang ada apa adanya seperti terlihat berikut ini:

1	2	3	4	5	6	7	8	9	10	11
no_bon	id_ksr	nama_ksr	no_polisi	id_warna	nama_warna	id_mrk	nama_merk	tahun	id_mekanik	nama_mekanik
05103214	ksr01	Dewi	B1234LB	w01	biru	m01	supra x	2005	SP	supriadi
05103218	ksr02	aura	B1235LB	w01	biru	m11	vario	2010	WN	wawan

12	13	14	15	16	17	18	19	20	21
kode_part	nama_part	tanggal	kuantum	harga	discount	jumlah	total	potongan	total_bayar
20W501000cc	oli federal 1000cc	25/10/2018	1	27000	1000	26000	50500	500	50000
serv001	service mesin		1	25000	500	24500			
20W501000cc	oli federal 1000cc	26/10/2018	2	27000	1000	52000	147500	1500	146000
serv001	service mesin		1	25000	500	24500			
kam001	kampas kopling		1	76000	5000	71000			

**Gambar 0.2** Relasi bon unnormalisasi

Pada hubungan diatas adalah merekam semua informasi saat ini direkam, data yang double tidak perlu ditulis. Terlihat baris/*record* yang kurang lengkap. Sulit untuk membayangkan seperti apa bentuk baris yang harus dibentuk untuk merekam informasi tersebut.

b. Bentuk normal pertama (1NF)

Bentuklah menjadi bentuk normal pertama dengan memisah-misahkan data pada atribut-atribut yang tepat dan bernilai atomik, juga seluruh *record*/baris harus lengkap adanya. Bentuk relasi adalah flat *file*. Dengan normal pertama kita dapat membuat satu tabel yang terdiri dari 21 atribut yaitu:

(no\_bon, id\_ksr, nama\_ksr, no\_polisi, id\_warna, nama\_warna, id\_merek, nama\_merek, tahun, id\_mekanik, nama\_mekanik, kode\_part, nama\_part, tanggal, kuantum, harga, discount, jumlah, total, potongan, total\_bayar).

Sehingga hasil daripada pembentukan normal pertama (1NF) adalah sebagai berikut:

1	2	3	4	5	6	7	8	9	10	11
no_bon	id_ksr	nama_ksr	no_polisi	id_warna	nama_warna	id_mrk	nama_merk	tahun	id_mekanik	nama_mekanik
05103214	ksr01	Dewi	B1234LB	w01	biru	m01	supra x	2005	SP	supriadi
05103214	ksr01	Dewi	B1234LB	w01	biru	m01	supra x	2005	SP	supriadi
05103218	ksr02	aura	B1235LB	w01	biru	m11	vario	2010	WN	wawan
05103218	ksr02	aura	B1235LB	w01	biru	m11	vario	2010	WN	wawan
05103218	ksr02	aura	B1235LB	w01	biru	m11	vario	2010	WN	wawan

12	13	14	15	16	17	18	19	20	21
kode_part	nama_part	tanggal	kuantum	harga	discount	jumlah	total	potongan	total_bayar
20W501000cc	oli federal 1000cc	25/10/2018	1	27000	1000	26000	50500	500	50000
serv001	service mesin	25/10/2018	1	25000	500	24500	50500	500	50000
20W501000cc	oli federal 1000cc	26/10/2018	2	27000	1000	52000	147500	1500	146000
serv001	service mesin	26/10/2018	1	25000	500	24500	147500	1500	146000
kam001	kampas kopling	26/10/2018	1	76000	5000	71000	147500	1500	146000

**Gambar 0.3** Relasi Bon memenuhi 1 NF



pada normal pertama tersebut masih terjadi banyak kelemahan terutama pada proses anomali insert, *update* dan *delete* berikut ini:

1) *Inserting* / penyisipan

Kita tidak dapat memasukan kode dan nama part baru tanpa adanya transaksi pembelian bon, sehingga part baru bisa dimasukan kalau ada transaksi pembelian bon tersebut.

2) *Deleting* / penghapusan

Bila suatu *record*/baris diatas dihapus, missal nomor bon 05103214 pada *record* pertama dihapus, maka akan berakibat pada penghapusan data seluruh *record* pada relasi tersebut padahal data tersebut masih diperlukan.

3) *Updating* / perubahan

Kode dan nama part terlihat ditulis berkali-kali, bila nama part berubah, maka setiap baris yang ada yang mengandung nama tersebut harus dirubah, bila tidak menjadi tidak konsisten.

Atribut jumlah total dan total\_bayar (merupakan atribut turunan) seharusnya tidak perlu, karena setiap harga x jumlah-potongan akan menghasilkan total, demikian untuk atribut total bayar, sehingga hasilnya akan lebih konsisten.

c. Bentuk normal kedua (2NF)

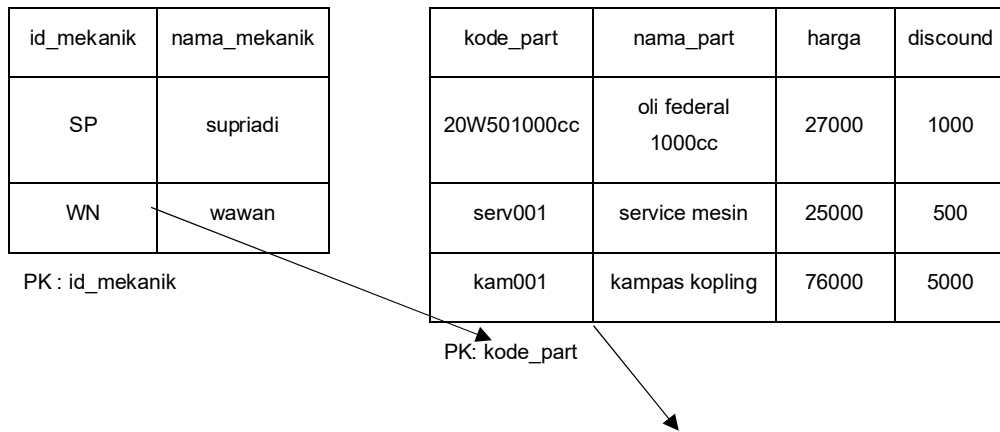
Bentuk normal kedua dengan melakukan dekomposisi relasi diatas menjadi beberapa relasi dan mencari kunci primer dari tiap tiap relasi tersebut dan atribut kunci haruslah unik

Melihat permasalahan Bon pembelian di atas, maka dapat diambil beberapa kunci kandidat: (No\_bon, no\_polisi, id\_mekanik, id\_ksr, kode\_part). Kunci kandidat tersebut nantinya bisa menjadi kunci primer pada relasi hasil dekomposisi.

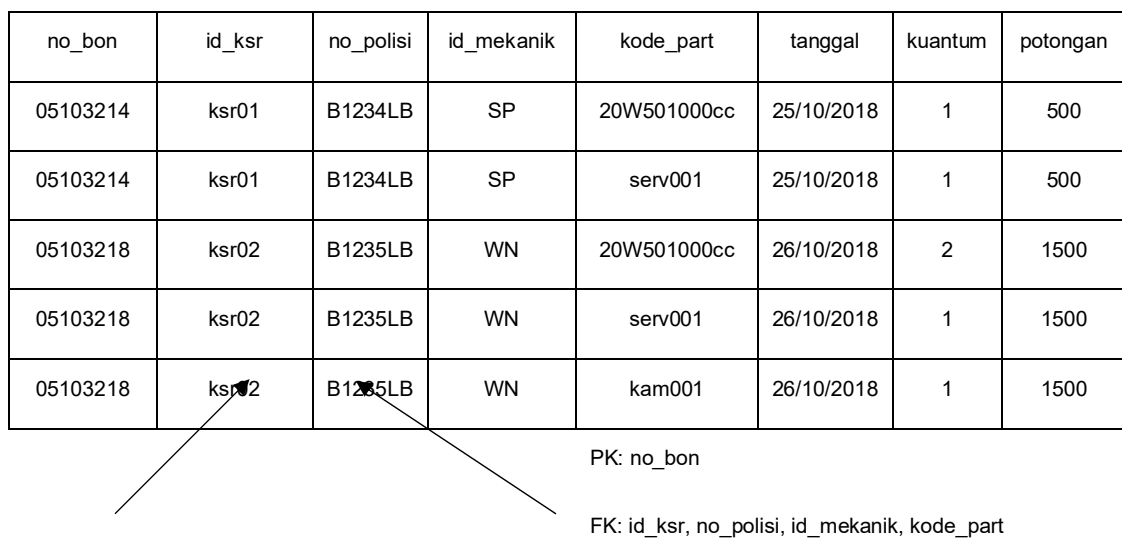
Dengan melihat normal pertama, kita dapat mendekomposisi menjadi 5 (lima) relasi beserta kunci primer yang ada yaitu: relasi kendaraan (no\_polisi), relasi mekanik (id\_mekanik), relasi kasir (id\_ksr), relasi part (kode\_part) dan relasi Bon (no\_bon). Dengan melihat ketergantungan fungsional atribut-atribut lain terhadap atribut kunci, maka didapatkan 5 (lima) relasi berikut:

mekanik

part



bon



kasir

id_ksr	nama_ksr
ksr01	Dewi
ksr02	aura

PK : id\_ksr

kendaraan

no_polisi	id_warna	nama_warna	id_mrk	nama_merk	tahun
B1234LB	w01	biru	m01	supra x	2005
B1235LB	w01	biru	m11	vario	2010

PK:  
no\_polisi**Gambar 0.4** Relasi bon memenuhi 2 NF

Dengan pemecahan relasi diatas, maka untuk pengujian bentuk normal kesatu (1NF) yaitu insert, *update*, dan *delete* akan terjawab. Kode dan nama part baru dapat masuk kapanpun tanpa adanya transaksi pada bon tersebut. Demikian pula proses *update* dan *delete* untuk tabel kasir, kendaraan,

mekanik,

pada bentuk normal kedua tersebut masih terjadi permasalahan yaitu pada relasi Bon, part dan kendaraan yaitu:

Atribut kuantum pada relasi bon, tidak tergantung pada kunci utama, atribut tersebut bergantung pada no\_bon + kode\_part, hal ini dinamakan ketergantungan transitif dan haruslah dipilah menjadi 2 relasi yaitu bon dan transaksi\_bon.

Atribut nama\_warna, nama\_merek, pada relasi kendaraan tidak tergantung pada kunci utama, atribut tersebut bergantung pada fungsi id\_warna dan id\_merek, hal ini dinamakan ketergantungan transitif dan haruslah dipilah 3 relasi yaitu kendaraan, warna dan merek.

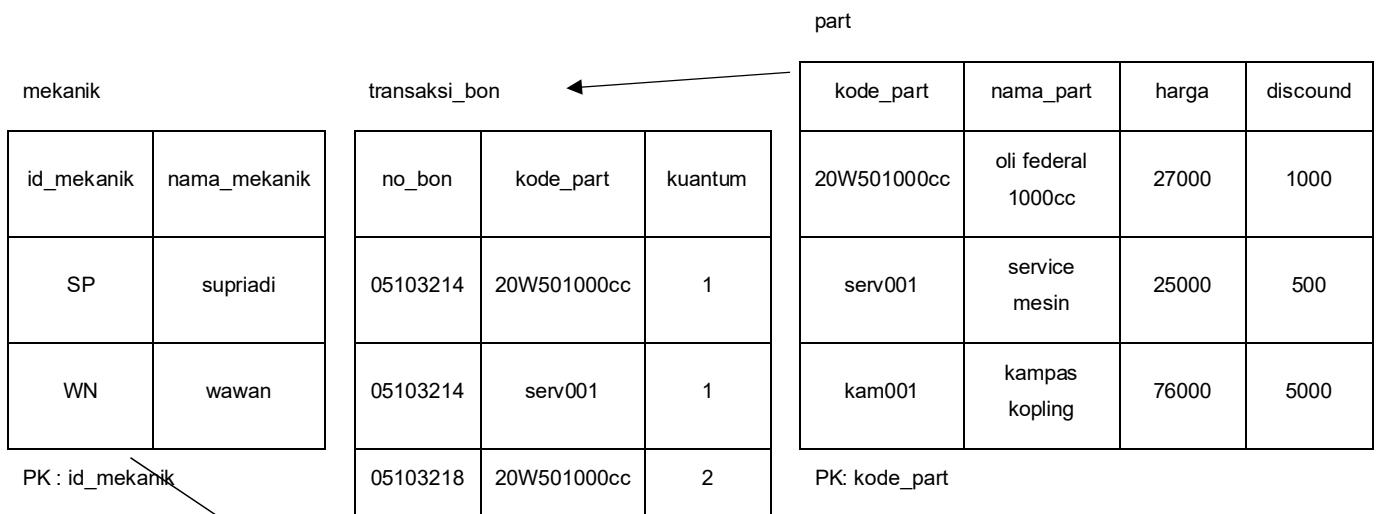
Masih terdapat satu pengulangan yaitu setiap kali satu bon yang terdiri dari lebih 1 part maka akan dituliskan no\_bon, id\_ksr, no\_polisi, id\_mekanik, tanggal. Hal ini harus dipisahkan menjadi 2 relasi masing masing bila terjadi penggandaan berulang-ulang.

d. Bentuk normal ketiga (3NF)

Bentuk normal ketiga mempunyai syarat, setiap relasi tidak mempunyai atribut yang bergantung transitif, harus bergantung penuh pada kunci utama dan harus memenuhi bentuk normal kedua (2NF).

Untuk memenuhi bentuk normal ketiga (3NF), maka pada relasi Bon harus didekomposisi (dipecah) lagi menjadi dua relasi Bon dan relasi

Transaksi\_Bon. Untuk kendaraan didekomposisi menjadi tiga relasi yaitu kendaraan, warna dan merek. Sehingga hasil 3NF adalah sebagai berikut:



05103218	serv001	1
05103218	kam001	1

PK/FK : no\_bon, kode\_part

bon

no_bon	id_ksr	no_polisi	id_mekanik	tanggal	potongan
05103214	ksr01	B1234LB	SP	25/10/2018	500
05103218	ksr02	B1235LB	WN	26/10/2018	1500

PK:

no\_bon

FK: id\_ksr, no\_polisi, id\_mekanik, kode\_part

warna

id_warna	nama_warna
w01	biru
w01	biru

PK :

id\_warna

merek

id_merek	nama_merk
m01	supra x
m11	vario

PK :

id\_merek

kasir

id_ksr	nama_ksr
ksr01	Dewi
ksr02	aura

PK :

id\_ksr

kendaraan

no_polisi	id_warna	id_merek	tahun
B1234LB	w01	m01	2005
B1235LB	w01	m11	2010

PK : no\_polisi

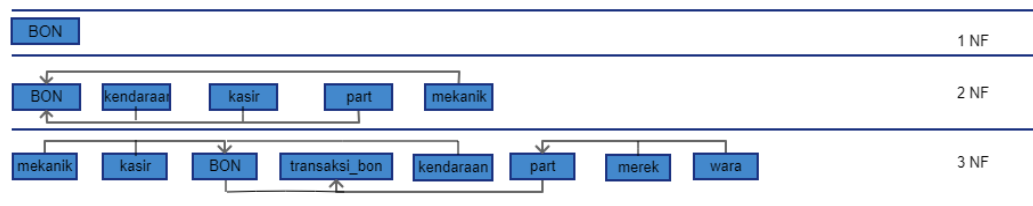
**Gambar 0.5** Relasi bon memenuhi 3 NF

*Primary key* pada relasi pada relasi mekanik adalah id\_mekanik, *primary key* pada relasi Bon adalah no\_bon dan *foreign key* id\_ksr, no\_polisi, id\_mekanik, *primary key* pada relasi Transaksi\_Bon adalah no\_bon +

kode\_part dan *foreign key* nya adalah kode\_bon + kode\_part, *primary key* dari relasi kasir adalah id\_ksr, *primary key* dari relasi kendaraan adalah no\_polisi dan *foreign key* id\_warna + id\_merek, *primary key* pada relasi merek adalah id\_merek dan *primary key* pada relasi warna adalah id\_warna.

e. Diagram dekomposisi

Kita dapat membuat diagram dekomposisi yang akan menjelaskan proses/tahapan uji normalisasi dari bentuk normal kesatu (1NF) sampai normal ketiga (3NF), seperti tampak pada gambar berikut:



**Gambar 0.6** Diagram dekomposisi relasi 1 NF sampai 3 NF

### C. SOAL LATIHAN/TUGAS

1. Jelaskan dan berikan contoh dari 1NF!
2. Jelaskan dan berikan contoh dari 2NF!
3. Jelaskan dan berikan contoh dari 3NF!
4. Carilah dokumen dasar seperti dalam contoh faktur diatas, kemudian lakukan Langkah normalisasi data, dari un-normalisasi hingga 3NF.

### D. REFERENSI

- Adyanata Lubis, S. M. (2016) BASIS DATA. Pasir Pengairan: DEEPUBLISH
- Didik setyadi, S. M. (2019) SISTEM BASIS DATA DAN SQL. Jakarta: Mitra Wacana Media.
- Kristanto, I. H. (1994). KONSEP DAN PERANCANGAN *DATABASE*. Yogyakarta: ANDI.

## GLOSARIUM

**Boyce-Codd Normal Form (BCNF)** Adalah koneksi dalam basis informasi yang harus direncanakan agar tidak memiliki ketergantungan yang tidak lengkap (halfway dependency), maupun ketergantungan yang bersifat transitif (transitif ketergantungan).

**Redundansi Data** adalah adanya informasi yang terlepas dari informasi asli dan memungkinkan perbaikan kesalahan dalam penyimpanan atau pengiriman informasi. Informasi tambahan dapat berupa duplikat lengkap dari informasi sebenarnya, atau pilih hanya bagian dari informasi yang memungkinkan lokasi kesalahan dan pembuatan kembali informasi yang tidak ada atau rusak.

**DBMS (*Database Management System*)** adalah kerangka kerja atau pemrograman yang secara eksplisit dimaksudkan untuk menangani basis informasi dan melakukan prosedur atas informasi yang ditentukan oleh berbagai klien.

**Entity** adalah "benda" atau "objek" dalam kenyataan yang dapat dikenali dari item yang tersisa.

**Micro Life Cycle** adalah siklus hidup basis informasi. Keduanya memiliki hubungan, khususnya pola keberadaan kerangka data hierarkis diidentifikasi dengan pola keberadaan kerangka kumpulan data yang menopangnya.

**Multivalue** adalah atribut yang menyimpan banyak nilai untuk setiap kemunculan tipe entitas..

**Update Anomaly** Itu adalah kesalahan atau blunder yang terjadi karena aktivitas perubahan *tuple/record* dari suatu koneksi.