

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.

Macam-macam anomali:

1. Anomali Peremajaan/Edit/Update

Anomali ini terjadi apabila terjadi pengubahan terhadap sejumlah data yang mubazir, tetapi tidak seluruhnya di ubah. Contoh:

PEMASOK	KOTA	BARANG	JUMLAH
Kartika	Jakarta	Monitor	10
Citra	Bandung	Disk Drive	20
Candra	Jakarta	Keyboard	25
Citra	Bandung	Mouse	30

Seandainya pemasok Citra berpindah ke kota lain, misalnya Bogor. Dan pengubahan hanya dilakukan pada data pertama (ada 2 buah data pemasok Citra), maka hasilnya sebagai berikut:

PEMASOK	KOTA	BARANG	JUMLAH
Kartika	Jakarta	Monitor	10
Citra	Bogor	Disk Drive	20
Candra	Jakarta	Keyboard	25
Citra	Bandung	Mouse	30

Terlihat ada ketidak konsistenan. Fakta pertama menyatakan bahwa pemasok citra berlokasi di Bogor, tetapi fakta kedua menyatakan di Bandung.

2. Anomali Penyisipan/insert

Anomali penyisipan terjadi jika pada saat penambahan hendak dilakukan ternyata ada elemen data yang masih kosong dan elemen tersebut justru menjadi kunci. Contoh:

KULIAH	RUANG	TEMPAT
Jaringan Komputer	Merapi	Gedung Utara
Pengantar Basis Data	Merbabu	Gedung Utara
Matematika I	Rama	Gedung Selatan
Sistem Pakar	Sinta	Gedung Selatan
Kecerdasan Buatan	Merapi	Gedung Utara

Masalahnya, bagaimana caranya menyimpan fakta bahwa RUANG baru bernama Arjuna terdapat pada Gedung Selatan? Penyisipan/penambahan

tidak dapat dilakukan mengingat tidak ada informasi KULIAH yang menggunakan ruangan tersebut.

3. Anomali Penghapusan/delete

Anomali penghapusan terjadi sekiranya sesuatu baris (tuple) yang tak terpakai dihapus dan sebagai akibatnya terdapat data lain yang akan hilang.

NO_SISWA	NAMA_KURSUS	BIAYA
10	Bahasa Inggris	60.000
10	Bahasa Perancis	80.000
10	Bahasa Mandarin	60.000
15	Bahasa Inggris	60.000
20	Bahasa Jepang	65.000

Contoh: apabila data siswa dengan identitas 20 yang mengambil kursus bahasa jepang dihapus, maka data yang menyatakan bahwa biaya kursus bahasa jepang sebesar 65.000 juga akan terhapus.

BENTUK-BENTUK NORMALISASI

- I. Bentuk Tidak Normal (Unnormalized Form)
Bentuk ini merupakan kumpulan data yang direkam, tidak ada keharusan mengikuti suatu format tertentu. Data dapat saja tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya.
- II. Bentuk Normal Kesatu (1NF/First Normal Form)
Bentuk normal kesatu mempunyai ciri yaitu setiap data dibentuk dalam flat file (file datar/rata), data dibentuk dalam satu record demi satu record dan nilai dari field berupa "Atomic Value". Tidak ada set atribut yang berulang-ulang atau atribut bernilai ganda (multivalue). Tiap field hanya satu pengertian, 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 memiliki sifat induknya.

Contoh 1

Entity : **Kelas**

Attribute : **Kode_Kelas, Nama_Kelas, Instruktur**

Kode_Kelas	Nama_Kelas	Instruktur
1234	Fisika Dasar	Suroso
1543	Matematika	Paulus
1775	Teknik Program	Bagus

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

Contoh 2

Entity : **Siswa**

Attribute : **No_Siswa, Nama, Wali_Studi, Kelas1, Kelas2, Kelas3**

No_Siswa	Nama	Wali_Studi	Kelas1	Kelas2	Kelas3
22890100	Tanzania	Zaman	1234	1543	
22890222	Nia	Rizki	1234	1775	1543

Siswa yang mempunyai No_Siswa, Nama, Wali_Studi mengikuti 3 mata pelajaran/kelas. Disini ada perulangan kelas 3 kali, ini bukan bentuk 1NF

Bentuk normal kesatu dari bentuk diatas adalah:

No_Siswa	Nama	Wali_Studi	Kode_Kelas
22890100	Tanzania	Zaman	1234
22890100	Tanzania	Zaman	1543
22890222	Nia	Rizki	1234
22890222	Nia	Rizki	1775
22890222	Nia	Rizki	1543

III. Bentuk Normal Kedua (2NF/Second Normal Form)

Bentuk normal kedua memiliki syarat yaitu bentuk data telah memenuhi kriteria bentuk normal kesatu. Attribute bukan kunci haruslah bergantung secara fungsi pada kunci utama/primary key. Sehingga untuk membentuk normal kedua haruslah sudah ditentukan kunci field. Kunci field haruslah unik dan dapat mewakili attribute lain yang menjadi anggotanya. Dari contoh relasi SISWA pada bentuk normal kesatu, terlihat bahwa kunci utama/primary key adalah nomor_siswa, Nama_siswa dan wali_studi bergantung fungsi kepada No_siswa, tetapi kode_kelas bukanlah fungsi dari SISWA maka file SISWA dipecah menjadi 2 relasi, yaitu:

Relasi SISWA

No_Siswa	Nama	Wali_Studi
22890100	Tanzania	Zaman
22890222	Nia	Rizki

Dan

Relasi AMBILKELAS

No_Siswa	Kode_Kelas
22890100	1234
22890100	1543
22890222	1234
22890222	1775
22890222	1543

IV. Bentuk Normal Ketiga (3NF/Third Normal Form)

Untuk menjadi bentuk normal ketiga maka relasi haruslah dalam bentuk normal kedua dan semua atribut bukan primer tidak punya hubungan yang transitif. Dengan kata lain setiap atribut bukan kunci haruslah bergantung hanya pada primary key dan pada primary key secara menyeluruh.

Contoh pada bentuk normal kedua diatas termasuk juga bentuk normal ketiga karena seluruh attribute yang ada disitu bergantung penuh pada kunci primernya.

- V. Boyce-Code Normal Form (BCNF)
- BCNF memiliki paksaan yang lebih kuat dari bentuk normal ketiga. Untuk menjadi BCNF, relasi haruslah dalam bentuk normal kesatu dan setiap atribut harus bergantung fungsi pada atribut superkey.
- Pada contoh dibawah ini terdapat relasi SEMINAR. Kunci primer adalah no_siswa+seminar. Siswa boleh mengambil satu atau dua seminar. Setiap seminar membutuhkan 2 instruktur, dan setiap siswa dibimbing oleh salah satu diantara 2 instruktur seminar tersebut. Setiap instruktur boleh hanya mengambil satu seminar saja.

Pada contoh dibawah ini No_Siswa dan seminar menunjukkan seorang instruktur.

Relasi SEMINAR

No_Siswa	Seminar	Instruktur
22890100	2281	Budi
22890101	2281	Kardi
22890102	2291	Mac
22890101	2291	Rahmat
22890109	2291	Rahmat

Bentuk relasi seminar adalah bentuk normal ketiga, tetapi tidak BCNF karena no seminar masih bergantung fungsi pada instruktur, jika setiap instruktur dapat mengajar hanya pada satu seminar. Seminar bergantung secara fungsi pada satu atribut bukan super key seperti yang di isyaratkan oleh BCNF. Maka relasi seminar haruslah dipecah menjadi dua yaitu:

Relasi PENGAJAR

Instruktur	Seminar
Budi	2281
Kardi	2281
Mac	2291
Rachmat	2291

Relasi SEMINAR-INSTRUKTUR

No Siswa	Instruktur
22890100	Budi
22890101	Kardi
22890102	Mac
22890101	Rachmat
22890109	Rachmat

Contoh Kasus

Proses perancangan database dapat dimulai dari dokumen dasar yang dipakai dalam sistem, contoh:

PT. TEMPO, TBK JALAN KHATIB SULAIMAN 60 PADANG			FAKTUR PEMBELIAN BARANG	
Kode Supplier : S02		Tanggal : 22/04/2005		
Nama Supplier : Hitachi		Nomor : 779		
KODE	NAMA BARANG	QTY	HARGA	JUMLAH
R02	RICE COOKER CC3	10	150.000	1.500.000
TOTAL FAKTUR				1.500.000
Jatuh Tempo Faktur : 26/05/2005				

PT. TEMPO, TBK JALAN KHATIB SULAIMAN 60 PADANG			FAKTUR PEMBELIAN BARANG	
Kode Supplier : G01		Tanggal : 24/04/2005		
Nama Supplier : Gobel Nusantara		Nomor : 998		
KODE	NAMA BARANG	QTY	HARGA	JUMLAH
A01	AC SPLIT ½ PK	10	1.350.000	13.500.000
A02	AC SPLIT 1 PK	5	2.000.000	10.000.000
TOTAL FAKTUR				23.500.000
Jatuh Tempo Faktur : 26/05/2005				

1. Step I bentuk Unnormalized

Bentuklah menjadi tabel unnormalized, dengan mencantumkan semua field yang ada. Tuliskan semua data yang akan direkam, bagian yang double tidak perlu dituliskan

Nofak	Kode Supp	Nama Supplier	Kode Brg	Nama Barang	Tanggal	Jatuh Tempo	QTY	Harga	Jumlah	Total
779	S02	HITACHI	R02	RICE COOKER CC3	22/04/2005	26/04/2005	10	150000	1500000	1500000
998	G01	GOBEL NUSANTARA	A01	AC SPLIT ½ PK	26/04/2005	26/04/2005	10	1350000	13500000	23500000
			A02	AC SPLIT 1 PK	26/04/2005	26/04/2005	5	2000000	10000000	23500000

2. Step II bentuk Normal Kesatu

Bentuklah menjadi bentuk normal kesatu dengan memisah-misahkan data pada field-field yang tepat dan bernilai atomic, juga seluruh record harus lengkap adanya. Bentuk file adalah flat file. Dengan bentuk normal kesatu ini telah dapat dibuat satu entity dengan 11 atribut yaitu:

Nofak	Kode Supp	Nama Supplier	Kode Brg	Nama Barang	Tanggal	Jatuh Tempo	QTY	Harga	Jumlah	Total
779	S02	HITACHI	R02	RICE COOKER CC3	22/04/2005	26/04/2005	10	150000	1500000	1500000
998	G01	GOBEL NUSANTARA	A01	AC SPLIT ½ PK	26/04/2005	26/04/2005	10	1350000	13500000	13500000
998	G01	GOBEL NUSANTARA	A02	AC SPLIT 1 PK	26/04/2005	26/04/2005	10	2000000	20000000	20000000

Namun bentuk normal kesatu ini masih banyak kelemahan yaitu:

a. Inserting / penyisipan

Kita tidak dapat memasukkan kode dan nama supplier saja tanpa ada transaksi pembelian, sehingga supplier baru dapat masuk apabila ada transaksi pembelian

b. Deleting / Penghapusan

Bila satu record diatas dihapus misalnya nomor faktur 779 maka berakibat pula penghapusan data supplier S02 (Hitachi), padahal data supplier tersebut masih dibutuhkan.

c. Updating / Pengubahan

Kode dan nama supplier terlihat ditulis berkali-kali, bila suatu ketika terjadi perubahan nama supplier misalnya maka harus mengganti disemua record yang mengandung hal tersebut. Bila ada yang terlewat maka membuat data tidak konsisten lagi.

d. Redudancy

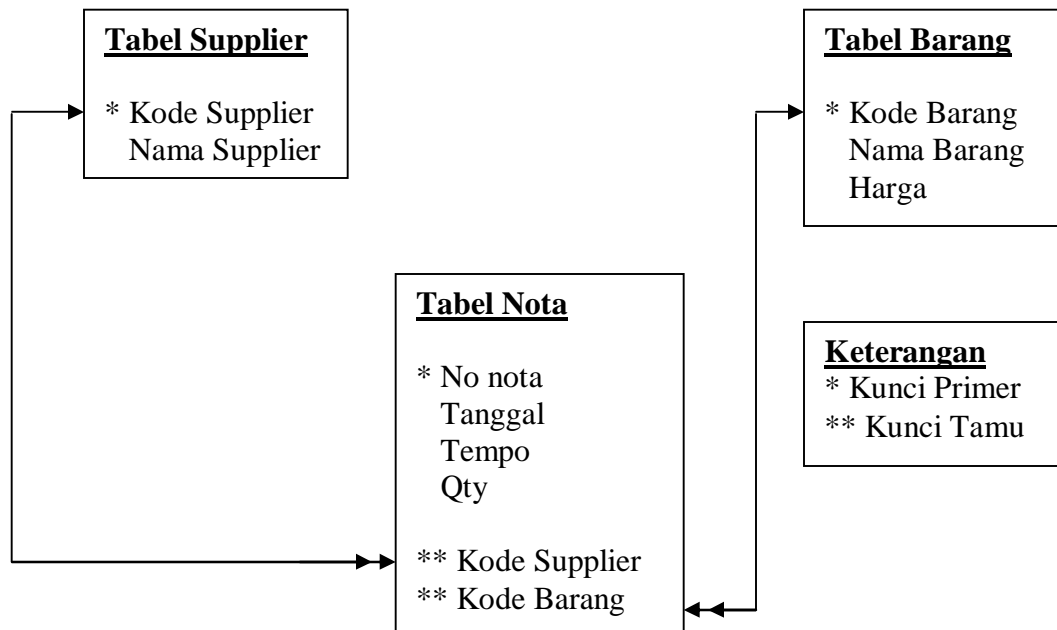
Field jumlah dan total diatas merupakan redudancy, karena setiap kali harga dikalikan dengan quantities akan menghasilkan jumlah. Maka field tersebut dapatlah dibuang, bila tidak dibuang maka mengakibatkan tidak konsisten. Tidak konsisten disini diakibatkan karena bila ada perubahan harga, hanya data harga yang diubah, data jumlah tidak maka nilai jumlah tidak sama dengan quantity kali harga.

3. Step III bentuk Normal Kedua

Pembentukan bentuk normal kedua dengan mencari kunci field yang dapat dipakai sebagai patokan dalam pencarian dan sifatnya unik. Melihat kondisi dari permasalahan faktur diatas dapat diambil kunci kandidat yaitu:

- no faktur
- kode supplier
- kode barang

bentuklah tiga tabel dengan kunci tersebut, lihatlah ketergantungan fungsional field-field lain terhadap field kunci, maka didapat tabel sbb:



Dengan pemecahan seperti diatas maka sebagian dari pertanyaan pengujian pada bentuk normal kesatu yaitu inserting, deleting, updating, dapat terjawab. Kode dan nama supplier baru dapat masuk kapanpun tanpa harus ada transaksi pada tabel nota, cukup dibuka tabel supplier dan disisipkan satu record baru. Demikian pula pada saat updating dan delete baik untuk tabel supplier dan juga tabel barang

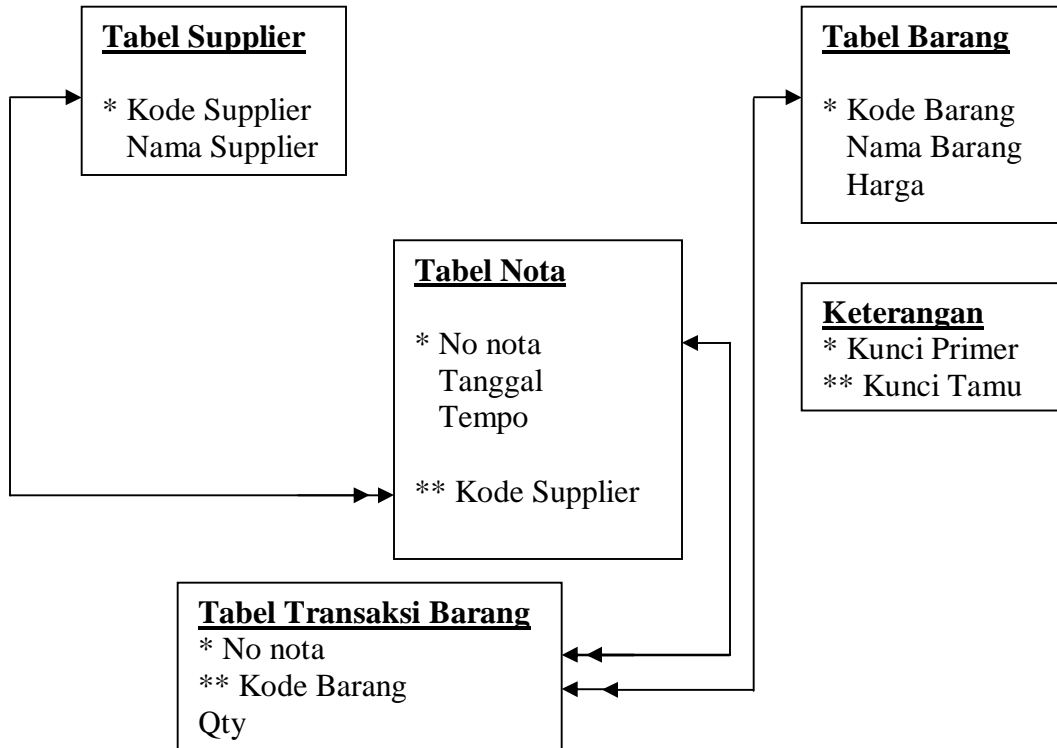
Namun permasalahan masih ada yaitu pada tabel nota:

- field qty pada tabel tersebut tidak bergantung penuh pada kunci primer nomor nota, ia juga bergantung pada kode barang. Hal ini disebut sebagai kebergantungan yang transitif dan haruslah dipisahkan dalam dua tabel.
- Masih terdapat redundancy yaitu setiap kali satu nota terdiri dari 5 macam barang yang dibeli maka 5 kalipula di tuliskan no nota, tanggal nota, tempo. Ini harus pula dipisahkan bila terjadi penggandaan tulisan berulang-ulang.

4. Step IV bentuk Normal Ketiga

Bentuk normal ketiga mempunyai syarat setiap tabel tidak mempunyai field yang bergantung transitif, harus bergantung penuh pada kunci utama.

Maka terbentuklah tabel sbb:



5. Step V Pengujian dengan Contoh

Pengujian disini untuk memastikan kebenaran isi tabel dan hubungan antara tabel tersebut. Ujilah bahwa setiap tabel haruslah punya hubungan dengan tabel yang lainnya. Bila tidak ada hubungan antar tabel maka dapat dikatakan perancangan untuk membuat satu database adalah gagal.

Tabel Supplier

Kode Supp	Nama Supplier
G01	Gobel Nusantara
S02	Hitachi

Tabel Barang

Kode Barang	Nama Barang	Harga
R01	Rice Cooker CC3	150000
A01	AC Split ½ PK	1350000
A02	AC Split 1 PK	2000000

Tabel Nota

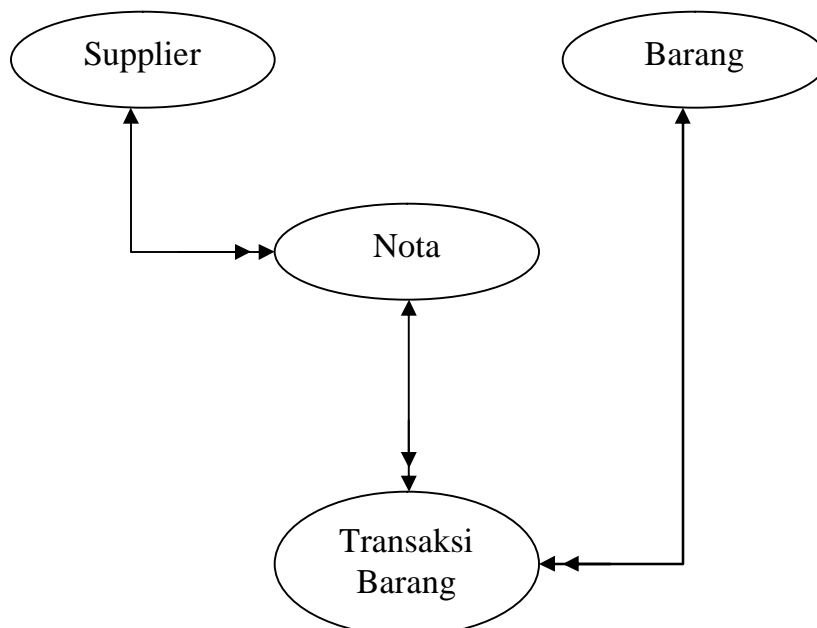
No Nota	Kode Supp	Tanggal	Tempo
779	S02	22/04/05	26/05/05
998	G01	22/04/05	26/05/05

Tabel transaksi

No Nota	Kode Barang	QTY
779	R01	10
998	A01	10
998	A02	10

6. Step VI Hubungan Relasi antar Tabel

Gambarkan hubungan relasi antar entity yang ada :



- a. satu supplier punya banyak nota
- b. satu nota punya beberapa transaksi barang
- c. satu barang terjadi beberapa kali transaksi pembelian

Sejauh mana Normalisasi Harus Dilakukan

Pertanyaan ini sering muncul atau dilontarkan oleh para mahasiswa dan praktisi-praktisi basis data. Tidak ada patokan dasar untuk itu. Pada dasarnya, kita harus terus melakukan langkah-langkah normalisasi sehingga kita tidak menemukan lagi anomali-anomali (baik anomali peremajaan, penyisipan dan penghapusan). Pada kenyataannya, normalisasi hingga ke 3NF seringkali sudah cukup memadai untuk menghilangkan anomali-anomali. Hanya kadang-kadang saja kita melangkah terus hingga ke BCNF dan 4NF, 5NF. Tetapi tidak ada standar baku sampai sejauh mana kita perlu melaksanakan langkah-langkah normalisasi. Pada prinsipnya lakukan normalisasi hingga anomali-anomali (peremajaan, penyisipan dan penghapusan) tidak ditemukan lagi.