

PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

ANISAH, M.KOM

HAL:1

STMIK ATMA LUHUR PANGKALPINANG DARTAR PUSTAKA

- Edhy Sutanta, "Basis Data dalam tinjauan konseptual, Andi Yogyakarta, 2011
- Dr.Said Mirza Pahlevi, "Tujuh Langkah Praktis Pembangunan Basis Data", PT. Elex Media Komputindi", 2013
- Indrajani, S.Kom, MM, "Pengantar dan Sistem Basis Data, PT.Elex Media Komputindi, 2011
- Indrajani, S.Kom, M.Kom, "Perancangan Basis Data Dalam Allin1", PT.Elex Media Komputindo, 2011
- Fathansyah, Basis Data, penerbit Informatika, 2011
- · Abraham Silberschatz, Henry F.Korth, S.Sudarshan, "Database System Concept: Third Edition, International Edition", The Mc Graw-Hill Companies, Inc, 2000.
- CJ Date HM, "An Introduction to Database System", Addison Wesley, 6th editions, 2001.
- Harianto Kristanto, "Konsep dan Perancangan Database", Andi Offset, Yogyakarta, 2000.
- · Linda Marlina, S.Kom, "Sistem Basis Data, Penerbit Andi, 2004
- Solichin Achmad, "MySQL 5 Dari Pemula Hingga Mahir", HTTP:
 //ACHATIM.NET, Universitas Budi Luhur Jakarta, 2010

HAL: 2

Pengertian Perancangan Basis Data/Database

 PERANCANGAN DATABASE adalah proses Pembuatan (develop) stuktur database sesuai dengan data yang dibutuhkan oleh user.

HAL:3

STMIK ATMA LUHUR PANGKALPINANG

ALASAN PERANCANGAN BASIS DATA

- Sistem basis data telah menjadi bagian dalam sistem informasi suatu organisasi
- Kebutuhan menyimpan data dalam jumlah besar semakin mendesak
- Fungsi-fungsi dalam organisasi semakin dikomputerisasikan
- Semakin kompleks data & aplikasi yg digunakan, maka relationship antar data harus dimodelisasikan
- Dibutuhkannya kemandirian data

HAL: 4

TUJUAN PERANCANGAN BASIS DATA

- untuk memenuhi kebutuhan-kebutuhan konten informasi dari pengguna dan aplikasi-aplikasi tertentu
- menyediakan struktur informasi yang alami dan mudah dipahami
- mendukung kebutuhan-kebutuhan pemrosesan dan objektifitas kinerja (waktu respon, waktu pemrosesan, dan ruang penyimpanan)

HAL:5

STMIK ATMA LUHUR PANGKALPINANG

Proses pembangunan Basis Data

Terdiri dari 2 Tahapan:

- 1. Tahapan Analisis dan Perancangan
- 2. Tahapan Implementasi

HAL: 6

Tahapan Analisa & Perancangan

Merupakan tahapan pemetaan atau pembuatan model dari dunia nyata menggunakan notasi perancangan basis data tertentu serta pembuatan deskripsi implementasi basis data.

dibagi menjadi 3 tahapan:

- perancangan basis data secara konsep(Conceptual Database Design)
- Perancangan basis data scara lojik(Logical Database Design)
- 3. Perancangan basis data secara fisik(Physical Database Design)

HAL:7

STMIK ATMA LUHUR PANGKALPINANG

PERANCANGAN BASIS DATA SECARA KONSEP(Conceptual Database Design)

- Merupakan proses pembuatan data model yang digunakan pada sebuah organisasi/perusahaan dan tidak bergantung pada seluruh aspek fisik basis data
- Data Model: sekumpulan konsep konsep yang mendeskripsikan data disuatu organisasi, menggambarkan hubungan antar data tersebut, dan memberikan batasan-batasan(Constraints) pada data.

HAL:8

Perancangan Basis Data Secara Konsep

Langkah 1 : penemuan dan analisis fakta

Meliputi:

- Identifikasi bagian organisasi yang akan didukung oleh aplikasi basis data(bagian yang akan menggunakan basis data
- identifikasi pemakai utama(siapa calon pemakai basis data, pejabat tertentu, staf tertentu)
- Penemuan fakta.
- 4. Rangkum hasil penemuan fakta.
- Output langkah ini adalah: dokumen yang berisikan penjelasan rinci tentang kebutuhan data, kebutuhan transaksi dan kebutuhan sistem dari pemakai basis data(dokumen Spesifikasi Kebutuhan pemakai)

HAL: 9

STMIK ATMA LUHUR PANGKALPINANG

Perancangan Basis Data Secara Konsep

Penemuan fakta

Metode yang digunakan bisa dengan menggunakan pengajian dokumen terkait, interview, observasi, penyebaran kuisioner.

Galilah informasi mengenai:

- 1. Sasaran dari misi pembangunan basis data(Mission objective)
- Kebutuhan data yang akan disimpan pada basis data(data requirements)
- 3. Kebutuhan transaksi terhadap data yang akan disimpan pada basis data(transaction requirements)
- 4. Kebutuhan sistem basis data yang akan dibuat(system requirements)
- Mencakup juga pengidentifikasian view pemakai.

HAL: 10

Perancangan Basis Data Secara Konsep

Rangkum Hasil Penemuan Fakta

Pendekatan yang bisa digunakan :

- pendekatan terpusat/centralize Approach(menggabungkan semua view pemakai menjadi satu view pemakai besar dan membangun data model(ERD terhadap pemakai besar tersebut).
- Pendekatan pengintegrasian view/view integration Approach(membangun data model lokal untuk masing-masing view pemakai, kemudian menggabung data model lokal menjadi satu data model global setelah melakukan normalisasi tabel.
- pendekatan gabungan/hybrid Approach(mengelompokkan view-view pemakai menjadi beberapa kelompok, kemudian gabung semua view pemakai kelompok. Kemudian membangun data model untuk masing-masing view pemakai kelompok. Menggabungkan data model view pemakai kelompok menjadi satu model data global setelah melakukan normalisasi tabel.

HAL: 11

STMIK ATMA LUHUR PANGKALPINANG

Perancangan Basis Data Secara Konsep

- Langkah 2: Membuat Diagram hubungan Entitas(ERD)
- Pembuatan ERD dilakukan berdasarkan dokumen spesifikasi kebutuhan pemakai yang merupakan output dari langkah sebelumnya

HAL: 12

Perancangan Basis Data Secara Lojik

Langkah 3: Memetakan ERD ke tabel

Meliputi:

- Petakan jenis entitas
- Petakan jenis hubungan entitas
- Petakan atribut bernilai jamak

Langkah 4 : Menormalkan Struktur Tabel Meliputi :

- Pemeriksaan normalitas tabel
- Normalisasi tabel
- Pemeriksaan validitas tabel

HAL: 13

STMIK ATMA LUHUR PANGKALPINANG

Perancangan Basis Data Secara Fisik

- Langkah 5 : Membuat Definisi Skema Basis Data
- · Meliputi:
- Pengodean dan kompresi tabel
- Memilih DBMS
- Mendefinisikan tabel dasar
- Menangani data turunan
- Langkah 6 : Merancang organisasi file & indeks
- · Meliputi:
- Menganalisa transaksi pemakai
- Menentukan organisasi file
- Menentukan indeks
- Membuat dokumentasi dan DDL indeks

HAL: 14

Tahapan Implementasi

- Merupakan tahapan pembuatan basis data pada DBMS menggunakan program aplikasi klien DBMS
- Tahapan 7 : mengimplementasikan DDL
- Meliputi:
- Menggunakan baris perintah
- Menggunakan antar muka grafis.

HAL: 15

STMIK ATMA LUHUR PANGKALPINANG

Perancangan Basis Data

- Untuk memahami perancangan basis data perlu mengenal daur hidup pengembangan sistem secara utuh.
- Perancangan basis data merupakan bagian dari tahapan perancangan sistem
- Tahapan perancangan sistem merupakan merupakan salah satu dari sejumlah tahapan yang ada pada daur hidup pengembangan sistem.

HAL: 16

Pengembangan sistem

- Ada beberapa pembagian tahapan dalam pengembangan sebuah sistem :
- 1. metodologi yang disebut waterfall atau air terjun membagi daur hidup pengembangan sistem menjadi 6 tahapan yaitu : konsepsi, pendahuluan, analisis, perancangan, implementasi, dan pengujian
- 2. Mcleod mengemukakan ada 4 tahapan yaitu: perencanaan, analisis, perancangan, dan implementasi
- 3. Fabbri dan Schwab membagi daur hidup pengembangan sistem menjadi 5 tahapan yaitu : Studi kelayakan, rencana pendahuluan, analisis sistem, perancangan sistem dan implementasi sistem

HAL: 17

STMIK ATMA LUHUR PANGKALPINANG

Tahapan Study Kelayakan

- Identifikasi terhadap kebutuhan sistem baru mulai dilakukan
- Identifikasi tidak hanya didasarkan oleh kebutuhan baru yang dihendaki manajemen(yang selama ini belum terpenuhi)
- Tetapi juga harus memperhatikan pada kebutuhan sistem yang sudah ada baik sistem manual maupun otomasi.

HAL: 18

Hasil Tahapan Study Kelayakan

- hasil tahapan study kelayakan dapat berupa :
- Daftar kebutuhan
- Perkiraan biaya untuk membuat sistem baru dan solusi yang dihendaki(didasarkan oleh DBMS yang digunakan(apakah oracle, Visual Foxpro DII, da juga bedasarkan pada komputer apa yang nantinya mau dipakai(apakah mainframe, minikomputer, atau mikrokomputer).

HAL: 19

STMIK ATMA LUHUR PANGKALPINANG

Tahapan rencana Pendahuluan

- Menentukan lingkup proyek atau sistem yang akan ditangani
- Hal ini digunakan untuk menentukan lingkup proyek

HAL: 20

Tahapan Analisa sistem

- Pada langkah ini dilakukan analisa data yang dibutuhkan, Penganalisaan ini dapat dilakukan secara langsung, yaitu dengan mendatangi langsung tempat atau objek yang dijadikan sistem implementasi.
- Proses analisa ini dapat dilakukan melalui wawancara atau dengan mencari data pada objek tujuan sehingga validasi data tercapai. Data-data yang valid tersebut siap diimplentasikan kedalan sistem database

HAL: 21

STMIK ATMA LUHUR PANGKALPINANG

Tahapan Perancangan Sistem

- Perancangan basis data
- Perancangan Basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna.
- meliputi : perancangan basis data secara konseptual, secara logis, dan secara fisis

HAL: 22

Tahapan Implementasi

- Setelah perancangan secara logika dan secara fisik lengkap, maka sistem basis data dapat diimplentasikan. Perintah-perintah dalam DDL dan SDL(storage definition language) dari DBMS yang dipilih dapat dikompilasi, dihimpun dan digunakan untuk membuat skema basis data dan file-file basis data (yang kosong).
- Jika data harus dirubah dari sistem komputer sebelumnya, perubahan-perubahan yang rutin mungkin diperlukan untuk format ulang datanya yang kemudian dimasukkan ke database yang baru.
- Spesifikasi secara konseptual diuji dan dihubungkan dengan kode program dengan perintah-perintah dari embedded DML yang telah ditulis dan diuji. Setelah transaksi- transaksi telah siap dan data telah dimasukkan ke dalam basis data, maka tahap perancangan dan implementasi telah selesai, maka pengoperasian sistem basis data dimulai.

HAL: 23



PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

Perancangan Konseptual Basis Data

HAL: 24

TEHNIK PENDEKATAN PERANCANGAN BASIS DATA

Dalam merancang basis data dapat dilakukan dengan dua buah tehnik pendekatan yaitu :

- 1. Menerapkan normalisasi terhadap struktur tabel yang telah diketahui.
- 2. Langsung membuat Model Entity Relationship.

HAL: 25

STMIK ATMA LUHUR PANGKALPINANG

TEHNIK PENDEKATAN PERANCANGAN BASIS DATA

Dalam pendekatan Normalisasi :

- 1. Perancang basis data bertitik tolak dari situasi yang nyata.
- 2. Perancang basis data telah memiliki item-item yang siap ditempatkan dalam baris dan kolom pada table.
- 3. Demikian juga dengan sejumlah aturan tentang keterhubungan antara item data.

Dalam pendekatan model Entity Relationship:

Langsung membuat model data lebih tepat dilakukan jika yang telah diketahui baru prinsip-prinsip sistem secara keseluruhan.

HAL: 26



PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

Normalisasi

HAL: 27

STMIK ATMA LUHUR PANGKALPINANG

PENGERTIAN NORMALISASI

Normalisasi adalah suatu teknik menstrukturkan/memecah/mendekomposisi data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data

Permasalahan yang dimaksud adalah adanya anomallies/Penyimpangan yang terjadi akibat adanya kerangkapan data dalam relasi dan inefisiensi pengolahan

HAL: 28

NORMALISASI

Basis data dikatakan normal jika:

- Relasi yang digunakan telah optimal
- Dilakukan pengujian berdasarkan kriteria bentuk normal
- Diwujudkan dengan normalisasi yaitu dekomposisi relasi menjadi relasi baru yang lebih sederhana

HAL: 29

STMIK ATMA LUHUR PANGKALPINANG

NORMALISASI DATA

Konsep dasar

- Normalisasi adalah metode untuk mengelompokkan atribut-atribut menjadi suatu relasi sehingga membentuk Well Structure Relation (struktur yang baik)
- 2. Normalisasi merupakan proses pengelompokkan elemen data menjadi tabel-tabel
- 3. Untuk menghindari duplikasi data
- 4. Merupakan proses mendekomposisikan relasi yang masih memiliki beberapa anomali untuk menghasilkan relasi yang lebih sederhana dan *well-structured*

HAL: 30

Well-Structured Relations

- Sebuah relasi (relasi) yang memiliki data redundancy yang minimal dan memungkinkan user untuk melakukan insert, delete, dan update baris (record) tanpa menyebabkan inkonsistensi data
- 2. Tujuannya untuk menghindari beberapa anomali:
 - Insertion Anomaly menambah record baru mempengaruhi user untuk membuat duplikasi data
 - Deletion Anomaly menghapus record mungkin menyebabkan hilangnya data yang akan dibutuhkan pada record lain
 - Modification Anomaly merubah data pada sebuah record mempengaruhi perubahan pada record lain karena adanya duplikasi.

HAL: 31

STMIK ATMA LUHUR PANGKALPINANG

NORMALISASI DATA

Sebelum mengenal lebih jauh mengenai normalisasi, ada beberapa konsep yang harus diketahui terlebih dahulu :

- 1. Field/ Atribut kunci (Key)
- 2. Ketergantungan Fungsional (Functional Depedency)

HAL: 32

Field/ Atribut kunci (Key)

Key adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (row) dalam tabel secara unik. Terdapat beberapa macam key yang dapat diterapkan pada suatu tabel, yaitu:

1. Super Key

 Satu atau lebih atribut (kumpulan atribut) yang dapat membedakan setiap baris data dalam sebuah tabel secara unik.

Pada tabel nasabah terdapat atribut-atribut sebagai berikut :

- no_rek, nama, no_ktp, tempat_lahir, tgl_lahir, alamat
- Super Key: no_rek: karena unik tidak mungkin ganda
- no_ktp : karena unik tidak mungkin ganda

HAL: 33

STMIK ATMA LUHUR PANGKALPINANG

Field/ Atribut kunci (Key)

- 2. Candidate Key
 - Merupakan kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah tabel secara unik.
- Primary Key Pada sebuah tabel dimungkinkan adanya lebih dari satu candidate key, salah satu dari candidate key (jika memang ada lebih dari satu) dapat dijadikan sebagai primary key.
- 4. Alternate Key Adalah candidate key yang tidak menjadi/ tidak dipakai sebagai primary key.
- 5. Foreign Key Foreign key (kunci tamu) adalah satu atribut yang melengkapi satu relationship yang menunjukan ke induknya.

HAL: 34

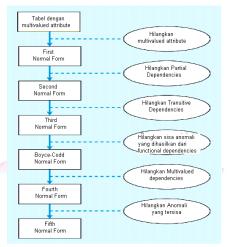
Functional Dependencies

- Normalisasi dilakukan berdasarkan analisa dari functional dependencies (relationship antar atribut)
- Functional Dependency: Nilai sebuah atribut (the determinant) menentukan nilai atribut yang lainnya.
- Setiap yang bukan key secara fungsional harus tergantung pada setiap candidate key (primary Key)

HAL: 35

STMIK ATMA LUHUR PANGKALPINANG

Langkah-langkah dalam Normalisasi



HAL: 36

First Normal Form(1NF)

- Sudah tidak ada repeating group
- Tidak memiliki multivalued attributes
- Setiap nilai atribut hanya mempunyai nilai tunggal

HAL: 37

STMIK ATMA LUHUR PANGKALPINANG

First Normal Form(1NF)

AMIK ATMA LUHUR

Laporan nilai semester genap 2008/2009

Nim	: 0611502342
Nama	: Agus Hando <mark>ko</mark>

Kode	Mata Kuliah	Nama Dosen	Lokasi	Nilai
K3C	SBD	Ari	5.1.1	A
P1A	Bahasa C	TIKA	4.1.1	C

HAL:38

AMIK ATMA LUHUR

Laporan nilai semester genap 2006/2007

Nim : 0611502350

Nama : Desi R

Kode	Mata Kuliah	Nama Dosen	Lokasi	Nilai
K1C	IMK	Usman	5.1.1	C
P1A	Bahasa C	TIKA	4.1.1	С

HAL: 39

STMIK ATMA LUHUR PANGKALPINANG

First Normal Form(1NF)

NIM	NAMA	KODE	MATA KULIAH	DOSEN	LOKASI	NILAI
06115002342	AGUS H	кзс	SBD	ARI	5.1.1	A
	The Control	P1A	Bahasa C	TIKA	4.1.1	С
0611502350	DESI R	K1C	IMK	USMAN	5.1.1	С
	74	P1A	Bahasa C	TIKA	4.1.1	С

RELASI YANG BELUM NORMAL TAHAP PERTAMA

HAL: 40

First Normal Form(1NF)

NIM	NAMA	KODE	MATA KULIAH	DOSEN	LOKASI	NILAI			
06115002342	AGUS H	кзс	SBD	ARI	5.1.1	A			
06115002342	AGUS H	P1A	Bahasa C	TIKA	4.1.1	С			
0611502350	DESI R	K1C	IMK	USMAN	5.1.1	С			
0611502350	DESI R	P1A	Bahasa C	TIKA	4.1.1	С			

RELASI YANG SUDAH NORMAL TAHAP PERTAMA

HAL: 41

STMIK ATMA LUHUR PANGKALPINANG

Second Normal Form (2NF)

- 1NF dan setiap atribut non-key sepenuhnya secara fungsional tergantung pada semua primary key.
 - Setiap atribut non-key harus didefinisikan oleh semua key, bukan oleh bagian dari key
 - Tidak memiliki partial functional dependencies(sebagian PK menentukan nonkey lainnya)
- relasi karyawan belum dalam 2nd Normal Form (2NF)

HAL: 42

Third Normal Form (3NF)

- dalam Third Normal Form (3NF) relasi harus dalam 2NF dan tidak ada transitive dependencies yang ada pada relasi.
- Transitive dependency adalah ketika ada atribut yang secara tidak langsung tergantung sama key dan atribut tsb tergantung pada atribut lain yang bukan key.

HAL: 43

STMIK ATMA LUHUR PANGKALPINANG

Third Normal Form (3NF)

- Relasi pada normal ke-3 sudah cukup pada kebanyakan praktek aplikasi database
- 2. Namun, 3NF tidak menjamin semua anomali sudah dihilangkan.
- 3. Untuk itu dilakukan beberapa normalisasi tambahan dilakukan untuk menghilangkan anomali anomali yang masih ada.

HAL: 44

Catatan

- Umumnya rancangan relasi dalam basis data telah optimal jika memenuhi kriteria bentuk 3NF
- Level normalisasi ditentukan berdasarkan kriteria bentuk normal, bukan banyaknya langkah menstruktur/dekomposisi/pemecahan sebuah relasi

HAL: 45

STMIK ATMA LUHUR PANGKALPINANG

Boyce-Codd Normal Form (BCNF)

 Boyce-Codd Normal Form (BCNF) Relasi harus sudah dalam bentuk ketiga dan setiap atribut harus bergantung fungsi pada atribut super key/ semua determinannya merupakan candidate key.

HAL: 46

Contoh

KARYAWAN

Apa primary key-nya?

Composite: Emp_ID, Course_Title

HAL: 47

STMIK ATMA LUHUR PANGKALPINANG

Anomali pada karyawan

- **Insertion** tidak dapat memasukkan data karyawan baru yang tidak mengambil kursus
- Deletion jika pegawai 400 dihapus, kita akan kehilangan informasi tentang keberadaan kelas Marketing
- Modification menaikan gaji pegawai 100 mengharuskan kita untuk meng-update beberapa records
 - Mengapa beberapa anomali ini muncul?
 Karena kita telah menyatukan 2 tema (entity) dalam satu relasi. Hal ini menyebabkan adanya duplikasi, dan ketergantungan antar entitas

HAL: 48

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Permasalahan:

- ✓ Penyimpangan terjadi akibat kerangkapan data
- ✓ Nilai rinci data dal<mark>am relasi memiliki keterga</mark>ntungan terhadap nilai rinci data yang <mark>lain</mark>

Cara mengatasi permasalahan:

- Melakukan dekomposisi / pemecahan data sesuai ke dalam bentuk yang efisien
- ✓ Mengetes/menguji ketergantungan relatif nilai rinci data terhadap PK
- Usahakan atribut non kunci/non key hanya bergantung pada atribut PK dan tidak memiliki ketergantungan pada atribut lainnya

•

HAL: 49

STMIK ATMA LUHUR PANGKALPINANG

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Ketergantungan Fungsional (Functionally Dependency/FD)

Jika nilai rinci data pada suatu atribut menentukan/mengimplikasikan nilai rinci data pada atribut lain

Contoh:

Sebuah atribut (Y) tergantung secara fungsional terhadap atribut lain (X) jika:

- ✓ Nilai X berkaitan dengan nilai pada Y
- Setiap record yang memiliki sembarang nilai X, selalu berhubungan dengan nilai Y yang sama

Notasi:

FD: R.X -> R.Y

Keterangan :

FD : Functionally Dependency

R : Nama Relasi

X : Atribut penentu (determinan), yaitu CK
 Y : Atribut yang bergantung (dependent)

HAL: 50

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Relasi Tour

No_Anggota	Nama_Anggota	Alamat_Lokal	Tujuan	Biaya	Tanggal	
3246	Erna	Jl. Mawar 10	Bali	500000	1-1-2002	
5498	Erni	Jl. Menur 20 Lombok		750000	2-2-2002	
8730	Irna	Jl. Melati 5	Surabaya	300000	3-3-2002	
6593	Arni	Jl. Mawar 20	Bali	550000	2-2-2002	

Dalam Relasi Tour maka FD dapat dituliskan sbb:

FD: R.X -> R.Y

FD: (Tour.No_Anggota, Tour.Tujuan, Tour.Tanggal) -> (Tour.Nama_Anggota, Tour. Alamat_Lokal, Tour.Biaya)

Penjelasan:

Nama_Anggota dan Alamat_Lokal adalah bergantung pada No_Anggota, sedang Biaya bergantung pada Tujuan dan Tanggal

HAL: 51

STMIK ATMA LUHUR PANGKALPINANG

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Ketergantungan Fungsional Penuh (Full Functionally Dependency/FFD)

Pada suatu kombinasi atribut jika FD pada suatu atribut dan tidak FD pada bagian lain dari kombinasi atribut

Contoh:

Suatu atribut Y mempunyai ketergantungan fungsional penuh terhadap atribut X

- √ Y functionally dependency terhadap X
- ✓ Y tidak functionally dependency terhadap bagian tertentu dari X

Notasi:

FFD: R.X -> R.Y

- Keterangan :
- FFD : Full Functionally Dependency
 R : Nama Relasi
- X : Atribut penentu (determinan), yaitu CK
 Y : Atribut yang bergantung (dependent)

HAL: 52

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Relasi Tour

No_Anggota	Nama_Anggota	Alamat_Lokal	Tujuan	Biaya	Tanggal
3246	Erna	Jl. Mawa <mark>r 10</mark>	Bali	500000	1-1-2002
5498	Erni	Jl. Menur 20	Lombok	750000	2-2-2002
8730	Irna	Jl. Melati 5	Surabaya	300000	3-3-2002
6593	Arni	Jl. Mawar 20	Bali	550000	2-2-2002

Dalam Relasi Tour maka FFD dapat dituliskan sbb:

FFD: R.X -> R.Y

FFD : (Tour.Tujuan, Tour.Tanggal) -> (Tour.Biaya)

FFD : (Tour.No_Anggota) -> (Tour.Nama_Anggota, Tour.Alamat_Lokal)

Penjelasan:

- ✓ Biaya tour hanya bergantung sepenuhnya pada Tujuan dan Tanggal & tidak bergantung pada siapa yang melakukan tour
- ✓ Nama_Anggota dan Alamat_Lokal hanya bergantung sepenuhnya pada No_Anggota

HAL: 53

STMIK ATMA LUHUR PANGKALPINANG

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Relasi Karyawan

NIK	Nama_Karyawan	Golongan_Gaji	Gaji_Pokok
01001	Feri	(IIA	600000
01002	Fira	IIIB	650000
01003	Fina	IIIA	600000
01004	Fita	IVA	800000
01005	Fani	IIIB	650000

Dalam Relasi Karyawan maka TDF dapat dituliskan sbb:

TDF: R.X -> R.Y -> R.Z

TDF : (Karyawan.NIK) -> (Karyawan.Golongan_Gaji) -> (Karyawan.Gaji_Pokok)

Penjelasan:

- ✓ Golongan_Gaji FD pada NIK, dan Gaji_Pokok FD pada Golongan_Gaji
- ✓ Jadi nilai rinci data pada atribut Gaji_Pokok (Z) bergantung pada Golongan_Gaji

HAL: 54

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Ketergantungan Total (Total Dependency/TD)

Suatu atribut Y mempunyai ketergantungan total jika:

✓ Y functionally dependency terhadap X

✓ X functionally dependency terhadap Y

Notasi:

TD : R.X < -> R.Y

Keterangan:

TD : Total Dependency
R : Nama Relasi

X : Atribut penentu (determinan), sekaligus bergantung pada Y Y : Atribut yang bergantung (dependent) sekaligus penentu pada X

HAL: 55

STMIK ATMA LUHUR PANGKALPINANG

KETERGANTUNGAN DATA (DATA DEPENDENCY)

Relasi Karyawan

NIK	Na <mark>ma_</mark> Karyawan	Golongan_ <mark>G</mark> aji	Gaji_Pokok
01001	Feri	IIIA	600000
01002	Fira	IIIB	650000
01003	Fina	IIIA	600000
01004	Fita	IVA	800000
01005	Fani	IIIB	650000

Dalam Relasi Karyawan maka TD dapat dituliskan sbb:

TD : R.X < -> R.Y

TD : (Karyawan.Golongan_Gaji) <-> (Karyawan.Gaji_Pokok)

HAL: 56

KETERGANTUNGAN DATA (DATA DEPENDENCY)

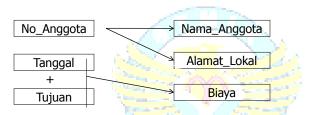


Diagram ketergantungan data dalam relasi Tour

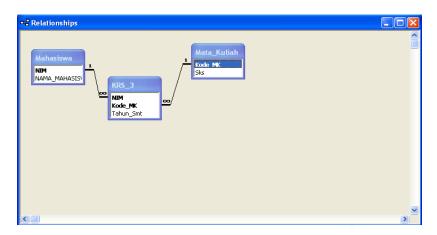


Diagram ketergantungan data dalam relasi Karyawan

57 HAL: 57

STMIK ATMA LUHUR PANGKALPINANG

NORMALISASI (NORMALIZATION)



HAL: 58

Catatan

- Tidak normal ke 1(terdapat repeating Group)
- Agar 1NF:Setiap Perpotongan baris dan kolom harus terisi
- Tidak Normal ke 2(terdapat Partial Dependency= sebagian PK menentukan nonkey lainnya
- Tidak Normal Ke 3(terdapat Transitif Dependency) yaitu : ada non key atribut menentukan atribu non key lainnya

HAL: 59



HAL: 60

							100			
noabsen	tglabsen	tapel	semester	kdkelas	nmkelas	nisn	nmsiswa	jmlsakit	jmlizin	jmlAlpa
AB001	12/02/2015	2014/2015	GENAP	KLS01	Х	1111	ALI	1	1	1
			100				100	100		
AB001	12/02/2015	2014/2015	GENAP	KLS01	Х	2222	BUDI	2	0	1
		100	100							
AB001	12/02/2015	2014/2015	GENAP	KLS01	X	3333	BADU	0	1	1
		100			0.00	70				

Tabel diatas sudah normal bentuk pertama(1NF) karena perpotongan baris dan kolomnya sudah terisi Akan tetapi belum normal bentuk ke dua karena masih ada Partial dependency.

HAL: 61

STMIK ATMA LUHUR PANGKALPINANG

Entity Relationship Model

- Entity-relationship (ER) data model adalah didasarkan pada suatu persepsi atas keadaan nyata di dunia yang terdiri dari sekumpulan obyek yang disebut entity dan relasi antara mereka
- Entity adalah suatu object yang dijelaskan oleh serangkaian atribut
- Ekspresi jumlah dari entity dengan entity yang lain dapat di-assosiasikan melalui himpunan relasi dinamakan mapping cardinality

HAL: 62

Entity Relationship Model

- Entity-relationship dikembangkan dalam rangka untuk memberikan fasilitas dalam perancangan database dengan memberikan kesempatan untuk membuat spesifikasi dari suatu skema yang merepresentasikan keseluruhan struktur logika dari database
- Model dituangkan dalam bentuk entity relationship diagram

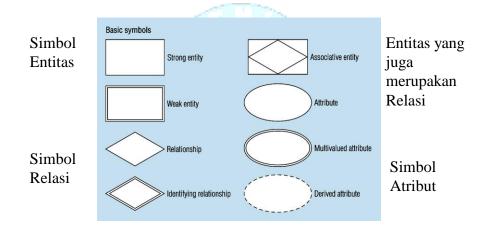
Komponen dalam model E-R

- Entity
- Relationship
- Attribute
- Mapping Cardinality

HAL: 63

STMIK ATMA LUHUR PANGKALPINANG

Simbol dalam ERD



HAL: 64

Entity/Entitas

Entitas adalah suatu object yang ada dan dapat dibedakan dengan obyek-obyek yang lain

Suatu entitas dapat *nyata*, misalnya seseorang, buku, dll Suatu entitas dapat berupa *abstrak*, misalnya suatu konsep, hari libur, dll

Suatu himpunan entitas (*entity set*) adalah suatu himpunan yang memiliki tipe yang sama

Suatu entitas direpresentasikan oleh suatu himpunan attribut

Secara formal, suatu atribut adalah suatu fungsi yang memetakan dari suatu himpunan entitas kedalam suatu domain

HAL : 65

STMIK ATMA LUHUR PANGKALPINANG

Entity/Entitas

Syarat sebuah Entitas:

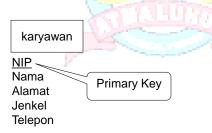
- Merupakan objek yang memiliki lebih dari satu entity instances (contoh) dalam database
 - Entity Instance untuk Entitas Mahasiswa adalah Rika, Andi, Della, dll
- Merupakan objek yang memiliki beberapa atribut.
- Bukan seorang user dari sistem.
- **Bukan** sebuah output dari sistem (contoh: laporan)
- Berinama dengan kata Benda

HAL: 66

Entity/Entitas

Strong entities

- Keberadaanya berdiri sendiri.
- Mempunyai Primary Key (unique identifier)
- Digambarkan dengan Persegi Empat dengan Garis Tunggal.



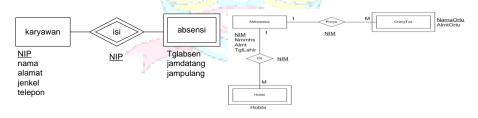
HAL: 67

STMIK ATMA LUHUR PANGKALPINANG

Entity/Entitas

Weak entity

- Tergantung pada strong entity. Tidak Dapat berdiri sendiri.
- Tidak mempunyai Primary Key (unique identifier)
- Digambar dengan dengan Persegi Empat dengan Garis double.



HAL: 68

Entity/Entitas

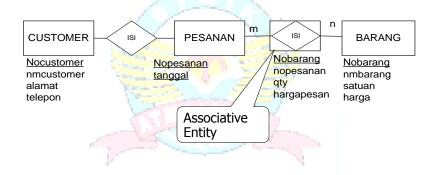
- Associative Entity
- Merupakan entity → yang mempunyai attributes
- Dan merupakan relationship → merupakan pengubung entities bersama.
- Kapan sebaiknya relationship dengan attributes menjadi sebuah associative entity?
 - Semua Relationships pada associative entity harus many
 - Associative entity bisa mempunyai arti tidak terikat pada Entity lain
 - Associative entity Lebih disukai mempunyai unique identifier, dan juga harus mempunyai attributes lain.
 - Ternary relationships harus dikonversi ke associative entities

HAL: 69

STMIK ATMA LUHUR PANGKALPINANG

Entity/Entitas

Contoh Associative Entity

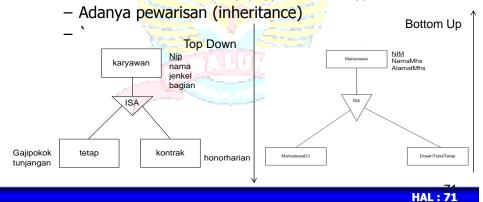


HAL: 70

Entity/Entitas

Generalisasi/Spesialisasi

- Pembedaan atas kekhususan satu atau beberapa entitas yang berjenis sama
- Membentuk entitas supertype dan subtype



STMIK ATMA LUHUR PANGKALPINANG

Entity/Entitas

Agregasi

- Meng-entitaskan sebuah relasi
- Terdapat data yang akan direlasikan dengan sebuah relasi.



By: ANISAH, M.KOM

HAL: 72

Attribute

- Sekumpulan attribut-attribut akan dapat menjelaskan suatu entitas
- Pertimbangkan himpunan entitas employee dengan attribut employee-name dan phone-number, sementara itu suatu telephone adalah suatu entitas yang memiliki attribut phone-number dan location
- Setiap attribut terdapat suatu himpunan nilai yang dapat diberikan pada attribut tersebut yang dikatakan sebagai domain dari suatu attribut
- Pembedaan pada seuah entitas digunakan suatu identitas dalam bentuk attribut kunci

HAL: 73

STMIK ATMA LUHUR PANGKALPINANG

Attribute

Memilih Nama Attribute

- Nama harus unik di dalam sistem
- Semua atribut yang menguraikan Entity atau Relationship tertentu harus diberi nama.
- Masing-Masing Relationship harus meliputi atribut yang menguraikan Entity tersebut dalam membentuk Ralationship.
- Nama penuh arti harus diterpilih sehingga E-R diagram adalah self-explanatory (menjelaskan isi dari dirinya)

Attribute

Klasifikasi Atribut

- Simple Attribute
 adalah atribut yang tidak dapat di breakdown menjadi beberapa komponen.
- Composite Attribute

 Adalah atribut yang dapat di breakdown menjadi beberapa komponen.
- Multivalued Attribute
 Adalah atribut yang memiliki lebih dari satu entity instance.
- Derived Attribute
 Adalah atribut yang merupakan nilai hasil perhitungan dari nilai atribut yang lain.

HAL: 75

STMIK ATMA LUHUR PANGKALPINANG

Relationship

- Suatu relasi adalah suatu assosiasi diantara beberapa entitas.
- Suatu himpunan relasi adalah suatu himpunan relasi yang memiliki tipe yang sama
- Jika e₁, e₂, e_n adalah himpunan entitas, maka suatu himpunan relasi r adalah suatu subset dari {(E₁, e₂, ..., e_n) | e₁ ∈ e₁, e₂ ∈ e₂, ..., e_n ∈ e_n} Dimana (e₁, e₂, ..., e_n) adalah suatu relasi

Relationship

Penamaan Relationship

- Berinama dengan "Kata Kerja"
- Tapi boleh juga menggunakan kata "Penghubung"
- Contoh:
 - Kata kerja → isi, tulis, kirim, dst
 - Kata penghubung→ ada, untuk, dalam, dst

HAL: 77

STMIK ATMA LUHUR PANGKALPINANG

Mapping Cardinality b_I b_2 \mathbf{a}_2 b₂ a_2 a₃ b₃ a_3 b_4 b₄ ONE TO ONE ONE TO MANY a_2 a_2 b_2 a_3 b_3 a_4 MANY TO ONE MANY TO MANY

IMPLEMENTASI BASIS DATA

Pengertian Implemetasi Basis Data:

Implementasi basis data merupakan upaya untuk membangun basis data fisik yang ditempatkan dalam memori sekunder (disk) dengan bantuan DBMS, tahap implementasi basis data diawali dengan melakukan transformasi dari model data yang telah selesai dibuat skema/ struktur basis data sesuai dengan DBMS yang dipilih.

Faktor-faktor yang mempengaruhi performansi sistem basis data:

- Kualitas dan bentuk perancangan basis data yaitu struktur basis data dan cara akses (algoritma aplikasi)
- Kualitas mesin
- Sistem Operasi
- DBMS (Data Base Management System)

HAL: 79

STMIK ATMA LUHUR PANGKALPINANG

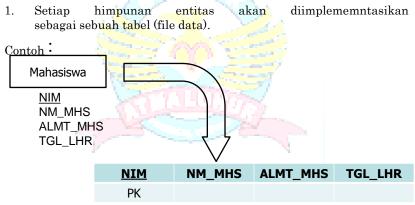
Tahap Implementasi Basis Data

- Transformasi dari model data yang telah selesai dibuat ke skema/struktur basis data yang sesuai dengan DBMS yang dipilih.
- Secara umum sebuah diagram ER akan direfresentasikan menjadi sebuah basis data secara fisik.

HAL: 80

TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

 Aturan umum dalam pemetaan model data yang digambarkan dengan Diagram ER menjadi Basis Data Fisik adalah sebagai berikut:

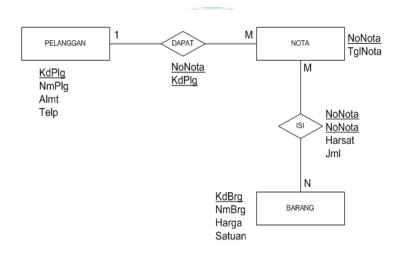


HAL: 81

STMIK ATMA LUHUR PANGKALPINANG

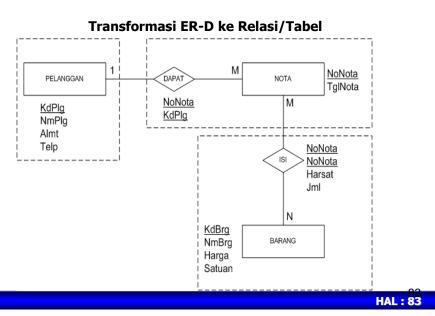
TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

Contoh Entity Relationship Diagram



HAL: 82

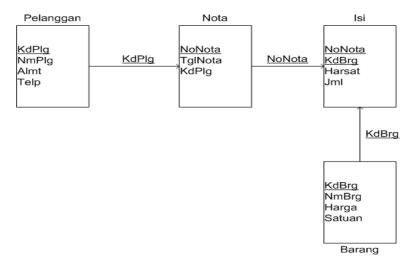
TRANSFORMASI MODEL DATA KE BASIS DATA FISIK



STMIK ATMA LUHUR PANGKALPINANG

TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

LRS



HAL: 84

TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

Tabel Pelanggan



HAL: 85

STMIK ATMA LUHUR PANGKALPINANG

Studi Kasus 1



<u>Asumsi :</u>

Pelanggan bisa mendapatkan lebih dari satu Nota

Berdasarkan data diatas Saudara diminta untuk membantu Toko Argya dalam pembuatan perancangan basis data dengan menggunakan Model ER :

- 1. buatlah ERD
- 2. Transformasi ERD ke LRS
- 3. LRS
- 4. Tabel

HAL: 86

Studi Kasus 2



Asumsi :

- Setiap Unit dapat membuat Lebih dari satu Form Permintaan Barang.
- Satu Form Permintaan Barang dapat membuat Lebih dari satu Bukti Penyerahan Barang, karena dapat dimungkinkan Jumlah yang diminta dapat tidak sama dengan jumlah yang diserahkan.

HAL: 87

STMIK ATMA LUHUR PANGKALPINANG

LATIHAN



KOPERASI BUDI LUHUR								
BUKTI PENARIKAN SIMPANAN								
NO. TARIK : NO. ANGGOTA : TGL. TARIK : NAMA ANGGOTA :								
NO NOMO SIMPAN		JUMLAH PENARIKAN						
1.								
3.								
	F	PETUGAS						
(X-20-X)								
,								

Asumsi :

- 1. Setiap Anggota dapat Memiliki Lebih dari satu Bukti Penyetoran simpanan dan satu bukti penyetoran simpanan dimiliki oleh satu anggota.
- 2. Setiap Anggota bisa mendapatkan Lebih dari satu Bukti Penarikan simpanan dan satu bukti Penarikan simpanan hanya didapat oleh satu anggota.

HAL:88



STMIK ATMA LUHUR PANGKALPINANG TRANSFORMASI MODEL DATA KE BASIS DATA FISIK MAHASISWA NAMA NIM ALAMAT TG-LAHIR JK NAMA FILE MEDIA ISI **ORGANISASI** PRIMARY KEY PANJANG RECORD JUMLAH RECORD STRUKTUR NAMA-FIELD **JENIS** LEBAR DESIMAL KETERANGAN

By: ANISAH, M.KOM

TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

NAMA FILE : NAMA RELASI

MEDIA : HARD-DISK / DISKET / TAPE / OPTICAL DISK (PILIH SATU)

: KETERANGAN TENTANG ISLFILE ISI

ORGANISASI : SEQUENTIAL / RANDOM / INDEX / INDEX SEQUENTIAL (PILIH SATU)

PRIMARY KEY : NAMA FIELD (BISA SATU ATAU LEBIH DARI SATU) PANJANG RECORD : TOTAL LEBAR FIELD (DALAM SATUAN BYTE)

JUMLAH RECORD : PERKIRAAN JUMLAH MAKSIMUM RECORD YANG DISIMPAN

STRUKTUR

NO	NAMA-FIELD	JENIS	LEBAR	DESIMAL	KETERANGAN
		CHARACTER	7		
		NUMERIC			
		DATE			<i>'</i>
		LOGICAL			
		DLL		2	



DIISI JUMLAH DIGIT DESIMAL YANG DIINGINKAN, & HANYA DIISI BILA JENIS FIELD **ADALAHNUMERIC**

DIISI PENJELASAN TENTANG ISI FIELD

HAL: 91

STMIK ATMA LUHUR PANGKALPINANG

TRANSFORMASI MODEL DATA KE BASIS DATA FISIK

CONTOH

: MAHASISWA NAMA FILE : HARDDISK MEDIA : DATA PRIBADI MAHASISWA ISI ORGANISASI : INDEX SEQUENTIAL PRIMARY KEY : NIM Jml record u/ file master =jml skrg + PANJANG RECORD : 69 BYTE (prakiraan pertambahan x umur sistem) 30.000 RECORD JUMLAH RECORD = 20<mark>000 mhs + (200</mark>0 mhs x 5 thn) **STRUKTUR** NO NAMA-FIELD LEBAR DESIMAL KETERANGAN **JENIS** NIM NUMERIC Nomor Induk 10 0 1 2 NAMA **ALPHABET** 20 Nama Mahasiswa 3 JK CHARACTER 1 Jenis Kelamin 4 TGL-LAHIR DATE 8 Tanggal Lahir 5 ALAMAT CHARACTER 30 Tempat Tinggal

HAL: 92

By: ANISAH, M.KOM

46



PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

PERTEMUA KE-6

Model Data Lanjutan

HAL: 93

STMIK ATMA LUHUR PANGKALPINANG

Model Data Lanjutan

Pengkodean Internal

Cara yang digunakan untuk menyatakan suatu data (atribut) dalam bentuk lain adalah untuk efisiensi ruang penyimpanan disebut dengan pengkodean (data coding).

Dari pemakaiann<mark>ya bisa dibedak</mark>an <mark>adanya</mark> pengkodean:

- Pengkodean eksternal (user define coding) : mewakili pengkodean yang telah digunakan secara terbuka dan dikenal oleh orang awam (ada pada fakta/ dunia nyata), contohnya: npm, kode_mtkul.
- Pengkodean internal (system coding) : kondisi sebaliknya dari pengkodean eksternal, contohnya : kode_dosen sebagai key alternatif. Pengkodean internal tidak hanya diterapkan pada pembuatan key alternatif, tapi juga dapat diterapkan pada atribut data lain (non key) yang memang kita kelola.

HAL: 94

MODEL DATA LANJUTAN

Ada tiga bentuk pengkodean yang dapat dipilih :

1. Sekuensial

Mengasosiasikan data dengan kode terurut, misalnya : (Sempurna, Baik, Cukup, Kurang, Buruk) dikodekan dengan : "A, B, C, D dan E".

2. Mnemonic

Membentuk suatu singkatan dari data yang ingin dikodekan, misalnya : jenis kelamin (Laki-Laki dan Perempuan) dikodekan dengan "L dan P ".

Blok

Pengkodean dinyatakan dalam format tertentu, misalnya No.Induk Mahasiswa dengan format XXYYYY

XX = dua digit terakhir angka tahun **masuk**

YYYY = nomor urut mahasiswa

HAL: 95

STMIK ATMA LUHUR PANGKALPINANG

MODEL DATA LANJUTAN

- DBMS dan Struktur Tabel
- Pembuatan berbagai tabel basis data adalah sebagai pekerjaan utama dalam tahap implementasi juga penetuan struktur dari tabel-tabel tersebut. Penentuan pilihan tipe data dan featur-featur tambahan untuk struktur table sangat bergantung pada DBMS yang dipilih.
- Pilihan-pilihan tipe data seperti:
- Data Angka
- a. Numerik : nilai angka dari tipe data ini menunjukan suatu jumlah misalnya : field sks, field semester, dll. Data numerik berupa :
 - Bilangan bulat (integer) terdiri dari Byte (1 byte),
 - · Small Integer (2 byte) dan Long Integer (4 byte).
 - Bilangan Nyata (real) terdiri dari Single (4 byte)
 - · bisa menampung hingga 7 digit pecahan dan Double
 - (8 byte) hingga 15 digit pecahan.

HAL: 96

MODEL DATA LANJUTAN

- DBMS dan Struktur Tabel (Lanjutan)
- 2. Data Teks (Character)
- Data teks ini dapat berupa abjad/ huruf, angka, karakter khusus atau gabungan dari ketiganya.
 Tipe ini tidak dapat melakukan operasi matematika.berupa:
 - a. Ukuran tetap (fixed character): data teks yang ukurannya pasti dan pendek, misalnya untuk field npm ukuran datanya selalu tetap.
 - b. Ukuran dinamis (variable character): data teks yang ukuran panjang pendeknya sangat bervariasi, misalnya untuk field nama_mhs dan alamat.

HAL: 97

STMIK ATMA LUHUR PANGKALPINANG

MODEL DATA LANJUTAN

- DBMS dan Struktur Tabel (Lanjutan)
- 2. Data Uang
- Jika DBMS yang digunakan menyediakan tipe data khusus untuk menyimpan data uang, misalnya bertipe money atau currency (8 byte).
- 4. Date / Tanggal
- Berupa gabungan angka dengan format tanggal tertentu (8 byte). Formatnya terdapat dua pilihan, diantaranya:
- day/month/year atau Hari/Bulan/Tahun
- month/day/year atau Bulan/Hari/Tahun

HAL: 98



PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

PERTEMUA KE-13

OBJECT ORIENTED DATABASE

(OODB)

HAL: 99

STMIK ATMA LUHUR PANGKALPINANG

KONSEP DASAR OODB

- ABSTRACTION
- OBJECT
- CLASSES
- MESSAGES
- ENCAPSULATION
- INHERITANCE
- POLYMORPHISM

HAL: 100

ABSTRACTION

- Proses mengidentifikasi aspek-aspek penting dari sebuah entitas dan mengabaikan atributatribut yang tidak penting.
- Berkonsentrasilah pada obyeknya, apa obyeknya dan apa yang dilakukan obyek itu sebelum memutuskan bagaimana mengimplementasikannya

HAL: 101

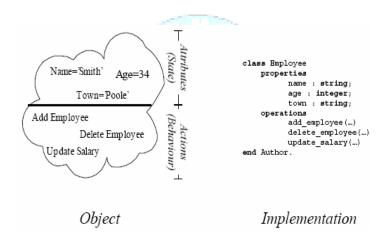
STMIK ATMA LUHUR PANGKALPINANG

OBJECT dan CLASS

- Definisi:
- "Object adalah sebuah representasi abstrak dari entitas dunia nyata yang mempunyai identitas tunggal, karakteristik, dan kemampuan untuk berinteraksi dengan objek-objek yang lain dan dirinya sendiri".
- "Object adalah entitas yang memiliki data yang menyatakan kondisi pada suatu saat dan sekumpulan operasi yang mengakses dan mengubah kondisi tersebut".
- Object biasanya Kata benda, bisa ada secara fisik (contoh mobil), dan bisa berupa konsep (contoh: pekerjaan).
- Sebuah object terdiri dari 2 bagian:
 - State: ditentukan oleh value/nilai dari atributnya
 - Behaviour: Sifat-2 object berdasarkan message yang diterima (ditentukan oleh operasi yang dilakukanya sendiri).
- CLASS merupakan Kumpulan objek-objek yang sejenis

HAL: 102

Contoh object dan implementasinya



HAL: 103

STMIK ATMA LUHUR PANGKALPINANG

METHOD

- Method Merupakan :
- Operasi (Actions) yang dapat dilakukan oleh sebuah object
 - Private methods, public method
- Komunikasi antar object dilakukan dengan memanggil methods yang ada pada object dengan menggunakan message

HAL: 104

MESSAGE

- Message merupakan :
- sarana komunikasi antar object.
 Message yang diterima akan mengaktifkan operasi operasi yang berada pada object

HAL: 105

STMIK ATMA LUHUR PANGKALPINANG

ENCAPSULATION

- Penggabungan data dan sejumlah operasi kedalam object
- Objek hanya bisa diakses melalui sekumpulan operasi yang telah didefinisikan.
- Disebut juga information hiding: penyembunyian informasi struktur internal objek
- Tidak berlaku pada basis data tradisional -> operasi select, insert, delete dan update bersifat generic.
- Pengguna objek hanya perlu memperhatikan interface objek yang mendefinisikan nama dan argumen setiap operasi.
- Bagian interface disebut signature, implementasi operasi disebut method.



HAL: 106

Keuntungan Encapsulation dan Information hiding:

- Data yang dienkapsulasi lebih terlindung
- Kita dapat membuat objek yang sangat handal dan konsisten karena object mempunyai kontrol penuh terhadap data
- Integritas lebih terjaga

HAL: 107

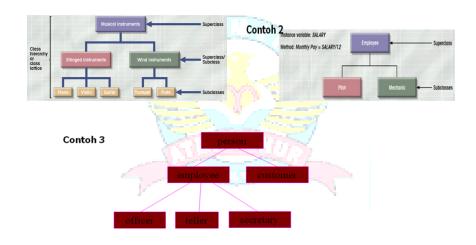
STMIK ATMA LUHUR PANGKALPINANG

CLASS HIERARCHIES DAN INHERITANCE

- Class dapat mempunyai 'anak'. Sebuah class dapat dibuat berdasarkan class lain.
- Konsep pewarisan: pendefinisian kelas baru yang mewarisi struktur dan operasi (behavior) dari kelas sebelumnya.
- Konsep ini mendukung pengembangan sistem secara incremental dan reusability dari kelas yang sudah ada.
- Class anak/turunan dapat dibuat untuk dapat mewarisi semua atribut dan perilaku dari class induk
- Class Hierarchies: Hubungan-hubungan antara Superclass dengan Subclass.

HAL: 108

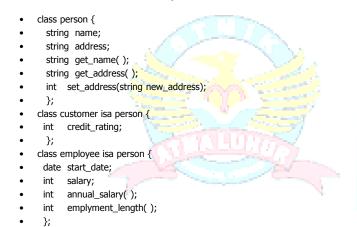
Contoh



HAL: 109

STMIK ATMA LUHUR PANGKALPINANG

Pseudecode untuk mendefinisikan Class hierarchy pada contoh 3:



HAL: 110

- int officer_number;
- };
- class teller isa employee {
- inthourly_per_week;
- intstation_number;
- };
- class secretary isa employee {
- inthourly_per_week;
- string manager;
- };
- Perhatikan bahwa method yang sudah didefinisikan pada superclass tidak ditulis lagi pada masing-masing subclass-nya, karena sudah otomatis.
- → hemat waktu programmer.

HAL: 111

STMIK ATMA LUHUR PANGKALPINANG

POLYMORPHISM

- Polymorphism: adalah kemampuan sebuah operasi suatu object pada object lain yang berbeda.
- Nama operasi sama, tetapi implementasinya berbeda tergantung dari tipe objeknya. Juga dikenal dengan METHOD OVERRIDE.
- Jika class mewarisi atribut dan perilaku dari class induk, maka bisa dimungkinkan perilaku dari class turunan berbeda dari class induknya.
- Dengan demikian untuk message yan sama mungkin akan memberikan efek yang berbeda tergantung class dari objek yang menerimanya.

HAL: 112

Contoh Polymorphism



HAL: 113



PERANCANGAN BASIS DATA

: 3 SKS

: PENGANTAR BASIS DATA

PERTEMUA KE-14

Relational Algebra dan SQL

HAL: 114

Query Language

RELATIONAL ALGEBRA (ALJABAR RELASIONAL)

Aljabar relasional (Relational Algebra) adalah relational operation yang digunakan untuk memanipulasi data pada basis data relasional. Aljabar relasional merupakan kumpulan operasi terhadap relasi di mana setiap operasi menggunakan satu atau lebih relasi untuk menghasilkan satu relasi yang baru. Aljabar relasional termasuk dalam kategori *bahasa prosedural* yang Menyediakan seperangkat operator untuk memanipulasi data.

Ada 2 operator dalam aljabar relasional, yaitu operator dasar yang fungsinya unik dan operator tambahan yang merupakan turunan dari satu atau lebih dari operator dasar dan mempunyai fungsi utama untuk menyederhanakan suatu ekspresi yang kompleks.

HAL: 115

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Operator Dasar:

- 1. Selection (σ)
- 2. Projection (π)
- 3. Cartesian Product (x)
- 4. Union (∪)
- 5. Difference (), dan
- 6. Renama (ρ)

Operator Tambahan:

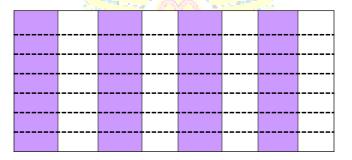
Intersection (\cap)

Semua operator diatas dapat diekspresikan ke dalam satu atau lebih tabel dan hasilnya berupa tabel

HAL: 116

SELECT (SELEKSI)

☐ Mengambil baris-baris (tuple) tertentu dari suatu tabel (relasi)



HAL: 117

STMIK ATMA LUHUR PANGKALPINANG

Query Language

TRANSFORMASI OPERATOR RELATIONAL ALGEBRA (ALJABAR RELASIONAL)
KE DALAM KASUS AKTUAL

Berikut akan dijelaskan contoh kasus dengan menggunakan tiga statement dan 3 tabel / relasi, yaitu :

Tiga Statement tersebut antara lain sebagai berikut :

- Natural Language (NL) merupakan bahasa ilmiah/sehari-hari yang digunakan oleh manusia dalam berkomunikasi dengan sistem
- Aljabar Relasional (AR) merupakan contoh penggunaan sintaks aljabar relasional dalam menjawab / menyelesaikan permintaan dari manusia (user) diatas
- 3. Tabel merupakan bentuk tampilan jawaban aljabar relasional dalam bentuk tabel/relasi dua dimensi yang terdiri dari kolom dan baris.

HAL: 118

Query Language

Berikut Tabel yang digunakan dalam aljabar relasional:

Tabel Pelanggan

No_Kartu	Nama	Jalan		Kota
22870	Agus	Cimone		Tangerang
48616	Jojo	Merdeka		Tangerang
03214	Fatah	Cikokol	AND THE PARTY NAMED IN	Tangerang Tangerang
37662	Jumali	Cimone		Ta ngerang

Tabel Rekening

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
771.225.421	Checking	3.000.000
315.222.310	Saving	4.000.000
342.256.010	Checking	6.000.000
511.333.279	Saving	1.000.000
122.003.007	Saving	8.500.000

Tabel Punya

No_Kartu	No_Rekening		
22870	342.256.010		
22870	511.333.279		
48616	771.225.421		
48616	315.222.310		
03214	012.145.002		
37662	122.003.007		

HAL: 119

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Definisi dan Contoh:

SELECTION (σ)

Operator ini menggunakan simbol σ (low-case omega). Operasi Selection menyeleksi tuple-tuple (record) pada sebuah relasi, yaitu tuple-tuple (record) yang memenuhi predicate syarat yang sudah ditentukan sebelumnya.

Selection operation disebut UNARY OPERATION karena beroperasi pada sebuah relasi. Komparasi dapat dilakukan dengan menggunakan : =, \neq atau <>, <, \leq , >, \geq pada predikat seleksi dan untuk menggabungkan serangkaian predikat adalah dengan menggunakan and (Λ) atau or (V)

Sintaks:

σ [karakteristik/kondisi] (tabel)

HAL: 120

Query Language

Contoh:

 NL: Cari semua tuple/record pada rekening yang saldonya lebih dari 4000000. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : σ Saldo > 4000000 (Rekening)

Tabel yang dihasilkan:

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
342.256.010	Checking	6.000.000
122.003.007	Saving	8.500.000

2. NL: Cari semua tuple/record pada rekening dengan tipe saving *dan* Saldonya lebih dari 4000000. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : σ Tipe = "Saving" Λ Saldo > 4000000 (Rekening)

Tabel yang dihasilkan:

No_Rekening	Tipe	Saldo
122.003.007	Saving	8.500.000

HAL: 121

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Contoh:

3. NL : Cari semua pelanggan yang tinggal di jalan Cimone. Bila diterjemahkan ke dalam aljabar relasional menjadi :

AR : σ Jalan = "Cimone" (Pelanggan)

Tabel yang dihasilkan:

Nama	No_Kartu	Jalan	Kota
Agus	22870	Cimone	Tangerang
Jumali	37662	Cimone	Tangerang

HAL: 122

Query Language

PROJECTION (π)

Operator ini menggunakan simbol π (phi). Operasi ini digunakan untuk memilih kolom (atrribut) dari tabel tertentu sesuai dengan sintaks.

Komparasi dapat dilakukan dengan menggunakan : =, ≠ atau <>, <, ≤, >, ≥ pada predikat seleksi dan untuk menggabungkan serangkaian predikat adalah dengan menggunakan and (Λ) atau or (V)

Sintaks:

 π A1, A2, A3,....An (tabel)

Keterangan:

 π (phi) adalah simbol operator Projection

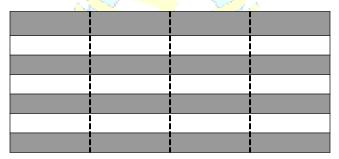
A1,A2,A3,... An adalah atribut yang dipilih (argument)

HAL: 123

STMIK ATMA LUHUR PANGKALPINANG

PROJECT (PROYEKSI)

☐ Mengambil kolom-kolom (atribut) tertentu dari suatu tabel (relasi)



HAL: 124

Query Language

Contoh:

 NL: Tampilkan semua nomor rekening dan saldo dari tabel rekening. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : π No_Rekening, Saldo (Rekening)

Tabel yang dihasilkan:

No_Rekening	Saldo
012.145.002	8.000.000
771.225.421	3.000.000
315.222.310	4.000.000
342.256.010	6.000.000
511.333.279	1.000.000
122.003.007	8.500.000

HAL: 125

STMIK ATMA LUHUR PANGKALPINANG

Query Language

 NL: Tampilkan semua no. rekening dan saldo dari semua rekening yang saldonya lebih dari 4000000. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : π No_Rekening, Saldo (σ Saldo > 4000000 (Rekening))

Tabel yang dihasilkan:

No_Rekening	Saldo		
012.145.002	8.000.000		
342.256.010	6.000.000		
122.003.007	8.500.000		

3. NL: Cari semua nomor rekening dan tipe dari semua rekening yang saldonya lebih besar dari 3000000 atau saldo lebih besar dari 5000000. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR: π No_Rekening, Tipe (σ Saldo > 3000000 V Saldo > 5000000 (Rekening))

Tabel yang dihasilkan:

No_Rekening	Tipe
012.145.002	Checking
315.222.310	Saving
342.256.010	Checking
122 003 007	Saving

HAL: 126

Query Language

CARTESIAN PRODUCT (x)

Operator ini menggunakan simbol x. Operator ini merupakan binary operation, yaitu operator yang beroperasi pada dua relasi.

Komparasi dapat dilakukan dengan menggunakan : =, \neq atau <>, <, \leq , >, \geq pada predikat seleksi dan untuk menggabungkan serangkaian predikat adalah dengan menggunakan and (Λ) atau or (V)

Sintaks:

R1 X R2

Keterangan:

- 1. R1 & R2 adalah relasi
- Operator Cartesian product akan merangkaikan setiap tuple dari R1 dan setiap tuple dari R2, sehingga jika R1 terdiri dari n-tuple dan R2 terdiri m-tuple, maka relasi R = R1 X R2 akan terdiri dari mn-tuple.

HAL: 127

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Contoh:

Bila R1 mempunyai arity K1 dan R2 mempunyai arity K2, maka R1 X R2 adalah himpunan tuple yang jumlah komponen pertamanya diambil dari tuple R1 dan K2 komponen terakhirnya <mark>diambil dari tuple R2, s</mark>ehingga arity dari relasi hasil Cartesian

Product adalah K1 + K2

В

b

а

b

Hasil operasi Cartesian Product

С

С

f

d

R1

а

d

С

R2

D E F
b g a
d a f

K = R1 X R2

Α	В	С	D	E	F
а	b	С	b	g	а
а	b	C	d	а	f
d	а	f	b	g	а
d	а	f	d	а	f
С	b	d	b	g	а
С	b	d	d	а	f

HAL: 128

Query Language

Contoh:

1. NL: Cari nomor kartu Pelanggan yang memiliki rekening = 511.333.279. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : π No_Kartu (σ No_Rekening = 511.333.279 ∧ Pelanggan.No_Kartu = Punya.No_Kartu (Pelanggan X Punya))

Langkah 1:

Tabel Pelanggan

No_Kartu	Nama	Jalan	Kota
22870	Agus	Cimone	Tangerang
48616	Jojo	Merdeka	Tangerang
03214	Fatah	Cikokol	Tangerang
37662	Jumali	Cimone	Tangerang

Tabel Punya

No_Kartu	No_Rekening
22870	342.256.010
22870	511.333.279
48616	771.225.421
48616	315.222.310
03214	012.145.002
37662	122.003.007

HAL: 129

STMIK ATMA LUHUR PANGKALPINANG

Langkah 2:

No_Kartu	Nama	Jalan	Kota	No_Kartu	No_Rekening
22870	Agus	Cimone	Tangerang	22870	342.256.010
22870	Agus	Cimone	Tangerang	22870	511.333.279
22870	Agus	Cimone	Tangerang	48616	771.225.421
22870	Agus	Cimone	Tangerang	48616	315.222.310
22870	Agus	Cimone	Tangerang	03214	012.145.002
22870	Agus	Cimone	Tangerang Tangerang	37662	122.003.007
48616	Jojo	Merdeka	Tangerang	22870	342.256.010
48616	Jojo	Merdeka	Tangerang	22870	511.333.279
48616	Jojo	Merdeka	Tangerang	48616	771.225.421
48616	Jojo	Merdeka	Tangerang	48616	315.222.310
48616	Jojo	Merdeka	Tangerang	03214	012.145.002
48616	Jojo	Merdeka	Tangerang	37662	122.003.007
03214	Fatah	Cikokol	Tangerang	22870	342.256.010
03214	Fatah	Cikokol	Tangerang	22870	511.333.279
03214	Fatah	Cikokol	Tangerang	48616	771.225.421
03214	Fatah	Cikokol	Tangerang	48616	315.222.310
03214	Fatah	Cikokol	Tangerang	03214	012.145.002
03214	Fatah	Cikokol	Tangerang	37662	122.003.007
37662	Jumali	Cimone	Tangerang	22870	342.256.010
37662	Jumali	Cimone	Tangerang	22870	511.333.279
37662	Jumali	Cimone	Tangerang	48616	771.225.421
37662	Jumali	Cimone	Tangerang	48616	315.222.310
37662	Jumali	Cimone	Tangerang	03214	012.145.002 1

Query Language

Langkah 3 (Hasil Akhir):

No_Kartu	Nama	Jalan	Kota	No_Kartu	No_Rekening
22870	Agus	Cimone	Tangerang	22870	511.333.279
48616	Jojo	Merdeka	Tangerang	22870	511.333.279
03214	Fatah	Cikokol	Tangerang	22870	511.333.279
37662	Jumali	Cimone	Tangerang	22870	511.333.279

HAL: 131

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Union (\cup)

Operator ini menggunakan simbol U. Operator ini merupakan binary operation, yaitu operator yang beroperasi pada dua relasi.

Komparasi dapat dilakukan dengan menggunakan : =, ≠ atau <>, <, ≤, >, ≥ pada predikat seleksi dan untuk menggabungkan serangkaian predikat adalah dengan menggunakan and (Λ) atau or (V)

Sintaks:

 $R = R1 \cup R2$

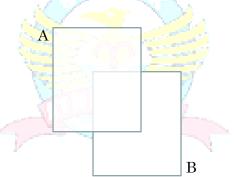
Keterangan:

- 1. R1 & R2 adalah relasi
- Operasi ini dapat dilaksanakan bila R1 dan R2 mempunyai arity yang sama, sehingga jumlah komponennya sama.
- 3. Domain dari atribut ke i dari R1 harus sama dengan domain dari atribut ke i dari R2 dimana i adalah bilangan integer

HAL: 132

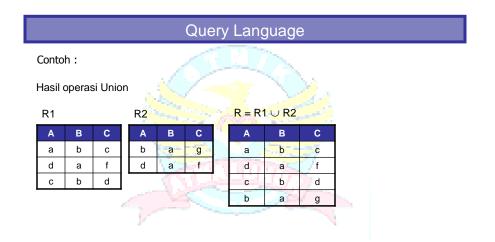
UNION (GABUNGAN)

☐ Membuat relasi baru, dari dua relasi lain, yang berisi semua *tuple* yang ada di relasi A, B dan keduanya



HAL: 133

STMIK ATMA LUHUR PANGKALPINANG



HAL: 134

Query Language

Contoh:

Tabel / Relasi DEPOSIT

Branch_Name	Account_Number	Customer_Name	Balance
Downtown	101	Johnson	500
Mianus	215	Smith	700
Perryridge	102	Hayes	400
Round Hill	305	Turner	350
Perryridge	201	Williams	900

Tabel / Relasi BORROW

Branch_Name	Loan_Number	Customer_Name	Amount
Downtown	17	Jones	1000
Redwood	23	Smith	2000
Perryridge	15	Hayes	1500
Pownal	29	Williams	1200
North Town	16	Adams	1300
Perryridge	25	Glenn	2500
Brighton	10	Brooks	2200

HAL: 135

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Contoh:

1. NL: Mencari nama Customer dari cabang Perryridge yang memiliki account atau loan atau kedua-duanya. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : π Customer_Name (σ Branch_Name = "Perryridge" (BORROW))

 π Customer_Name (σ Branch_Name = "Perryridge" (DEPOSIT))

Tabel yang dihasilkan:

Customer_Name
Hayes
Williams
Glenn

HAL: 136

Query Language

Berikut Tabel yang digunakan dalam aljabar relasional:

Tabel Pelanggan

No_Kartu	Nama	Jalan	Kota
22870	Agus	Cimone	Tangerang
48616	Jojo	Merdeka	
03214	Fatah	Cikokol	Tangerang
37662	Jumali	Cimone	Tangerang

Tabel Rekening

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
771.225.421	Checking	3.000.000
315.222.310	Saving	4.000.000
342.256.010	Checking	6.000.000
511.333.279	Saving	1.000.000
122.003.007	Saving	8.500.000

HAL: 137

STMIK ATMA LUHUR PANGKALPINANG

Query Language

DIFFERENCE (--)

Operator ini menggunakan simbol — . Operator ini merupakan binary operation, yaitu operator yang beroperasi pada dua relasi. Operator ini berfungsi untuk mengeliminasi entity / record dari suatu tabel yang ada pada tabel lain, dan kedua tabel harus memiliki atribut yang sama.

Sintaks:

[tabel 1] — [tabel 2] atau R = R1 – R2

Keterangan:

- 1. R, R1 & R2 adalah relasi
- 2. Relasi R1 dan R2 harus mempunyai arity yang sama, yaitu mempunyai jumlah atribut yang sama.

HAL: 138

Query Language

Contoh:

 NL: Mencari semua Pelanggan yang saldonya diatas 4000000 dan tidak bersifat saving. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR : π No_Kartu (σ tipe = "saving" (Rekening x Pelanggan))

 π No_Kartu (σ tipe = "checking" (Rekening x Pelanggan))

Tabel yang dihasilkan:

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
342.256.010	Checking	6.000.000

HAL: 139

STMIK ATMA LUHUR PANGKALPINANG

Query Language

RENAMA / RENAME (ρ)

Operator ini menggunakan simbol ρ . Operator ini berfungsi menyalin tabel lama menjadi tabel dengan nama baru.

Sintaks:

ρ [Ne<mark>w Name]^[Old Name]</mark>

Keterangan:

- 1. New Name = nama relasi baru
- 2. Old Name = nama relasi lama

HAL: 140

Query Language

Contoh:

 NL: Salin tabel Rekening dengan nama X. Bila diterjemahkan ke dalam aljabar relasional menjadi:

 $\mathsf{AR}: \rho \ \mathsf{X}^{\ (\mathsf{Rekening})}$

Tabel yang dihasilkan:

Tabel X

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
771.225.421	Checking	3.000.000
315.222.310	Saving	4.000.000
342.256.010	Checking	6.000.000
511.333.279	Saving	1.000.000
122.003.007	Saving	8.500.000

HAL: 141

STMIK ATMA LUHUR PANGKALPINANG

Query Language

INTERSECTION (∩)

Operator ini menggunakan simbol

Operator ini termasuk kedalam kategori operator tambahan, karena operator ini dapat diderivasi dari operator dasar.

Sintaks:

$$R = R1 \cap R2$$

Keterangan:

- 1. R1 adalah relasi dengan elemen E1, E2,...., En
- 2. R2 adalah relasi dengan elemen T1, T2,...., Tn
- 3. Menghasilkan relasi R dengan elemen yang terdapat di R1 juga terdapat di R2

HAL: 142

Query Language Contoh: Hasil operasi Intersection $R = R1 \cap R2$ R1 R2 Α В С Α С В b b d а С a g a d f d f а а С b d

HAL: 143

STMIK ATMA LUHUR PANGKALPINANG

Query Language

Contoh:

 NL: Mencari nomor kartu yang memiliki rekening bertipe saving. Bila diterjemahkan ke dalam aljabar relasional menjadi:

AR: π No_Kartu (σ tipe = "saving" Λ Rekening.No_Rekening = Punya.No_Rekening (Rekening × Punya)) $\cap \pi$ No_Kartu (σ tipe = "saving" Λ Rekening.No_Rekening = Punya.No_Rekening (Rekening × Punya))

Tabel yang dihasilkan:

Tabel Rekening

No_Rekening	Tipe	Saldo
012.145.002	Checking	8.000.000
771.225.421	Checking	3.000.000
315.222.310	Saving	4.000.000
342.256.010	Checking	6.000.000
511.333.279	Saving	1.000.000
122.003.007	Saving	8.500.000

Tabel Punya

No_Kartu	No_Rekening
22870	342.256.010
22870	511.333.279
48616	771.225.421
48616	315.222.310
03214	012.145.002
37662	122.003.007

HAL: 144

Query Language

Tabel yang dihasilkan:

No_Kartu
22870
48616
37662

HAL: 145

Tabel Nota Tabel Baran

STMIK ATMA LUHUR PANGKALPINANG

LATIHAN ALJABAR RELATIONAL

Tabel Barang

KdBrg	Nmbarang	satuan	harga
B0001	Komputer	unit	Rp.5.000.000,-
B0002	Mouse	buah	Rp.100.000,-
B0003	Printer	unit	Rp.600.000,-
B0004	FlashDisk	buah	Rp.250.000,-
B0005	HardDisk External	buah 🤼	Rp.1.000.000,-

Tabel ISI

NoNota	KdBrg	Jumlah
N0001	B0001	4
N0001	B0002	6
N0001	B0003	4
N0002	B0001	1
N0002	B0003	10

Tabel Nota

NoNota	TglNota
N0001	10/10/2010
N0002	12/10/2010

Pertanyaan : Buatlah Aljabar Relational dan hasilnya dari pertanyaan di bawah ini ;

- 1. Cari semua tuple/record pada barang yang harganya lebih kecil dari Rp.1.000.000
- 2. Tampilkan semua kode barang dan harga barang dari tabel barang
- 3. Tampilkan semua kode barang dan harga barang dari semua data barang yang harganya lebih besar dari Rp.500.000,-
- Cari semua kode barang dan harga barang dari semua barang yang harganya lebih besar dari Rp.800.000,- dan lebih kecil dari Rp.2.000.000,-
- 5. Cari nomor nota yang memiliki kode barang B0003.
- 6. Salin tabel isi dengan nama punya.

HAL: 146

STRUKTUR QUERY LANGUANGE

- Sebagai data sublanguange (DSL) terdiri atas:
- □ Data Definition Language (DDL)
 - Create Table
 - Drop Table
 - Create Index
 - Alter Table
 - Create View
 - Drop View
- ☐ Data Manipulation Language (DML)
 - Select
 - Update
 - Insert
 - Delete

HAL: 147

STMIK ATMA LUHUR PANGKALPINANG

MEMBUAT DATABASE BARU

☐ Bentuk umum membuat data base baru

Create database database name>

contoh: create database penjualan

☐ Bentuk umum mengaktifkan database

use <database name>

contoh: use penjualan

HAL: 148

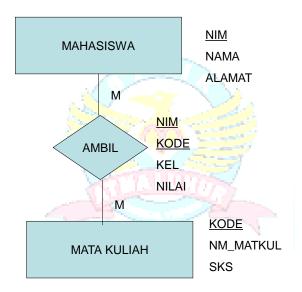
MEMBUAT RELASI BARU

☐ Bentuk umum Create Table
Create Table base-table
(column-definition
[,column-definition]...
[,primary-key-definition]
[,foreign-key-definition]...]])

Untuk 'column-definition' mempunyai bentuk : column-name data-type [NOT NULL]

HAL: 149

STMIK ATMA LUHUR PANGKALPINANG



HAL: 150

Tabel MAHASISWA

NIM	NAMA	ALAMAT
0311500012	AHMAD	JAKARTA
0322500023	RINA	CILEDUG
0322500045	ANI	JAKARTA

Tabel AMBIL

NIM	KODE	KEL	NILAI
0311500012	KP124	AA	80
0322500023	KP124	AA	50
0311500012	KP125	AB	60

Tabel MATA KULIAH

KODE	NM_MATKUL	SKS
KP124	SBD 1	2
KP125	SBD 2	3
KP126	PBD	3

HAL: 151

STMIK ATMA LUHUR PANGKALPINANG

Contoh CREATE TABLE

CREATE TABLE MAHASISWA (NIM CHAR(10) PRIMARY KEY NOT NULL,

NAMA CHAR(40));

CREATE TABLE AMBIL

(NIM CHAR(10) REFERENCES

MAHASISWA,

KODE CHAR(5) REFERENCES MATKUL,

KEL CHAR(2), NILAI INTEGER);

CREATE TABLE MATKUL (KODE CHAR(5) PRIMARY KEY NOT NULL, NM_MATKUL CHAR(40),

SKS INTEGER);

HAL: 152

MENAMBAH ATRIBUT

Bentuk umum Alter Table

Alter Table base-table

Add column data type;

Contoh:

ALTER TABLE S ADD DISCOUNT SMALLINT;

HAL: 153

STMIK ATMA LUHUR PANGKALPINANG

Mysql

Jenis Tipe Data MySQL

Jenis Tipe Data	Bentuk Penulisan	Ukuran	
TINYINT	TINYINT [(M)] [UNSIGNED] [ZEROFILL]	Bilangan bulat dengan kisaran nilai : SIGNED: -128 s/d 127 (-2 ⁷ s/d 2 ⁷ -1) UNSIGNED: 0 s/d 255 (0 s/d 2 ⁸ – 1)	1 byte
SMALLINT	SMALLINT [(M)] [UNSIGNED] [ZEROFILL]	Bilangan bulat dengan kisaran nilai : SIGNED: -32768 s/d 3276 7 (-2 ¹⁵ s/d 2 ¹⁵ -1) UNSIGNED: 0 s/d 65535 (0 s/d 2 ¹⁶ – 1)	2 byte
MEDIUMINT	MEDIUMINT [(M)] [UNSIGNED] [ZEROFILL]	Bilangan bulat dengan kisaran nilai : SIGNED: -8388608 s/d 8388607 (-2 ²³ s/d 2 ²³ -1) UNSIGNED: 0 s/d 1677215 (0 s/d 2 ²⁴ - 1)	3 byte
INT	INT [(M)] [UNSIGNED] [ZEROFILL]	Bilangan bulat dengan kisaran nilai : SIGNED: -2147683648 s/d 2147683647 (-2 ³¹ s/d 2 ³² -1) UNSIGNED: 0 s/d 4294967295 (0 s/d 2 ³² – 1)	4 byte

HAL: 154

Jenis Tipe Data MySQL

Jenis Tipe Data	Bentuk Penulisan	Keterangan	Ukuran
BIGINT	BIGINT [(M)] [UNSIGNED] [ZEROFILL]	Bilangan bulat dengan kisaran nilai: SIGNED -9223372036854775808 s/d 9223372036854775807 (-2 ⁶³ s/d 2 ⁶³ -1) UNSIGNED : 0 s/d 18446744073709551615 (0 s/d 2 ⁶⁴ – 1)	8 byte
FLOAT	FLOAT [(M,D)] [UNSIGNED] [ZEROFILL] atau FLOAT(4)	Bilangan pecahan presisi tunggal (single precision) dengan kisaran nilai: nilai minimum (tanpa nilai nol) ±1.175494351E-38 Nilai maksimum (tanpa nilai nol) ±6.402823466E+38	4 byte
DOUBLE	DOUBLE [(M,D)] [UNSIGNED] [ZEROFILL] atau FLOAT(8)	Bilangan pecahan presisi ganda (double precision) dengan kisaran nilai: nilai minimum (tanpa nilai nol) ±2.2250738580720E-308 Nilai maksimum (tanpa nilai nol) ±1.7976931348623157E+308	8 byte
DECIMAL	Decimal (M,D)	Bilangan desimal, dengan nilai tergantung pada besaran M dan D	M byte
CHAR	Char(M)	Data string dengan lebar data yang tetap (M). Maksimum lebar adalah 255	M byte
VARCHAR	Varchar(M)	Data string dengan lebar data yang bervariasi (M). Maksimum lebar adalah 255	M byte
DATE	-	Digunakan untuk tanggal dengan format penulisan "YYYY-MM- DD". Batasan nilai dari "1000-01-01 s/d "9999-12-31"	3 byte

HAL: 155

STMIK ATMA LUHUR PANGKALPINANG

MENGHAPUS ATRIBUT

Bentuk umum Alter Table

Alter Table base-table

DROP column data type;

Contoh:

ALTER TABLE S DROP DISCOUNT SMALLINT;

HAL: 156

MENGHAPUS RELASI

Bentuk umum DROP TABLE

DROP TABLE base-table

Contoh:

DROP TABLE S;

HAL: 157

STMIK ATMA LUHUR PANGKALPINANG

Bentuk umum DROP TABLE

DROP database <database name>

Contoh:

DROP database penjualan

HAL: 158

DATA MANIPULATION LANGUANGE (DML)

□ SELECT

Menampilkan sebagian atau seluruh isi dari suatu tabel / Menampilkan kombinasi isi dari beberapa tabel

□ UPDATE

Mengubah isi satu atau beberapa atribut dari suatu tabel

□ INSERT

Menambah satu atau beberapa baris nilai baru ke dalam suatu tabel

□ DELETE

Menghapus sebagian atau seluruh isi dari suatu tabel

HAL: 159

STMIK ATMA LUHUR PANGKALPINANG

SELECT

Perintah select digunakan untuk memilih dan menampilkan data dari suatu database, baik dari satu tabel atau lebih. Fungsi select dalam bentuk sederhana hanya mempunyai pasangan kata from yang berarti memilih tabel yang akan diselect

Bentuk umum perintah SELECT

SELECT < nama Kolom>

From <nama tabel>

Contoh: Select nama_pelanggan from pelanggan;

Menggunakan distinct

Select distinct < nama kolom >

From <nama tabel>

Contoh: Select distinct nama_pelanggan from pelanggan;

HAL: 160

SELECT (lanjutan...)

Selain from pelengkap dari fungsi select adalah where.

Where berfungsi untuk memberikan kondisi pemilihan sehingga hasil perintah select sesuai dengan yang dibutuhkan Operator relasiona dengan tanda =, <>, <, >, <=, >= Serta Operasi logika AND, or, xor

HAL: 161

STMIK ATMA LUHUR PANGKALPINANG

UPDATE

Update merupakan perintah untuk mengubah data yang sudahada di dalam tabel.

Bentuk umum update:

Update <tabel>

Set <nama kolom> = <value>

Where <kondisi>

Contoh: Update penjualan set total = 1000 where no_nota=101;

HAL: 162

INSERT

Insert merupakan sintaks umum yang digunakan untuk pengisian data ke dalam tabel

Bentuk umum:

Insert into nama_tabel values (daftar nilai), (daftar nilai),....

Contoh: Insert into Pelanggan ("yana", "pangkalpinang", "433577")

Bentuk umum yang lain:

Insert into nama_tabel (daftar kolom) values (daftar nilai),....

Contoh: Insert into mahasiswa (NIM, NmMhs) values ("1044300010", "Mustajab");

HAL: 163

STMIK ATMA LUHUR PANGKALPINANG

INSERT (lanjutan....)

Bentuk umum yang lain:

Insert into nama_tabel set nama_kolom = nilai

Contoh: Insert into mahasiswa set NIM ="1044300010",

NmMhs = "Mustajab";

HAL: 164

DELETE

Delete digunakan untuk menghapus record suatu tabel, sedangkan untuk menghapus data pada kolom tertentu menggunakan perintah update.

Bentuk umum:

Delete from <tabel>

Where <kondisi>

Contoh:

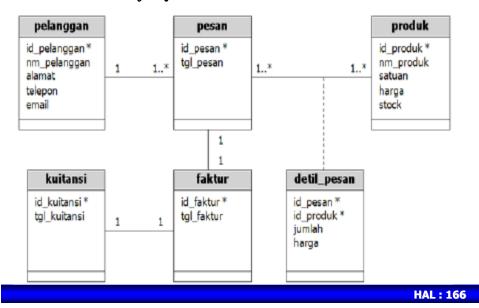
Delete from pelanggan

where alamat = "Sungailiat"

HAL: 165

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN



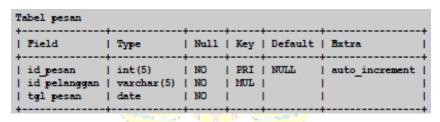
PERINTAH MySQL LANJUTAN

Tabel pelanggan		4			
Field	Туре	Null	Key	Default	Extra
id pelanggan		-	PRI		i
nm pelanggan alamat	varchar(40) text	I NO I NO			
telepon					
email	varchar(50)	I NO	I		
†		+	-+		

HAL: 167

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN



A								
Field	Туре	į	Null	į	Кеу	Ī	Default	Extra
harga		i I I	NO NO NO NO		PRI	į	0	

HAL: 168

PERINTAH MySQL LANJUTAN

Field	Туре	Null	Key	Default	Extra
id pesan	int(5)	NO	PRI		
id produk	varchar(5)	NO	PRI	1 1	
jumlah	int(5)	NO		10 1	
harga	decimal(10,0)	NO		10 1	

Tabel faktur					
Field	Туре	Null	Key	Default	Extra
•	int(5) int(5)	NO NO			auto increment

HAL: 169

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN

Pabel kuitansi		1			
Field	Туре	Null	Key	Default	Extra
id_kuitansi id_faktur tgl kuitansi	int(5)	NO	PRI	 MATT	auto_increment

HAL: 170

85

PERINTAH MySQL LANJUTAN

PERINTAH SELECT DENGAN JOIN

Untuk menggabungkan 2 (dua) atau lebih tabel, dapat menggunakan bentuk perintah JOIN.

1. Inner Join

Dengan inn<mark>er join tabel akan</mark> digabungkan dua arah, sehingga tidak ada data yang NULL di satu sisi.

Cara 1 menggunakan WHERE Bentuk Umum:

SELECT Tabel1.*, Tabel2.* FROM Tabel1, Tabel2 WHERE Tabel1.PK=Tabel2.FK;

HAL: 171

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN

INNER JOIN (LANJUTAN)

Contoh: Menggabungkan tabel pelanggan dan pesan dimana dalam tabel Pelanggan hanya menampilkan id, nama.

Perintah:

SELECT pelanggan.id_pelanggan, pelanggan.nm_pelanggan, pesan.id_pesan, pesan.tgl_pesan
FROM pelanggan, pesan
WHERE pelanggan.id_pelanggan=pesan.id_pelanggan

HAL: 172

HAL: 173

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN

INNER JOIN (LANJUTAN)

Hasilnya:

11asiiiiya		L	
id_pelanggan	nm_pelanggan	id_pesan	tgl_pesan
P0001	Achmad Solichin	1	2008-02-02
P0001	Achmad Solichin	5	2007-12-14
P0002	Budianto	2	2008-02-05
P0002	Budianto	3	2008-02-10
P0004	Amin Riyadi	4	2008-01-20
+		+	+

HAL: 174

PERINTAH MySQL LANJUTAN

INNER JOIN (LANJUTAN)

Cara 2 Menggunakan INNER JOIN

Bentuk Umum:

SELECT Tabel1.*, Tabel2.*

FROM Tabel INNER JOIN Tabel 2

ON Tabel1.PK=Tabel2.FK;

Perintah menggabungkan tabel pelanggan dan pesan:

SELECT pelanggan.id_pelanggan,

pelanggan.nm_pelanggan, pesan.id_pesan,

pesan.tgl_pesan

FROM pelanggan INNER JOIN pesan

ON pelanggan.id_pelanggan=pesan.id_pelanggan;

HAL: 175

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN

INNER JOIN (LANJUTAN)

Hasilnya:

+		+	
id_pelanggan	nm_pelanggan	id_pesan	tgl_pesan
P0001 P0001 P0002 P0002 P0004	Achmad Solichin Achmad Solichin Budianto Budianto Amin Riyadi	5 2 3	2008-02-02 2007-12-14 2008-02-05 2008-02-10 2008-01-20

HAL: 176

PERINTAH MySQL LANJUTAN PERINTAH SELECT DENGAN JOIN (LANJUTAN)

2. Outer Join

Dengan outer join, tabel akan digabungkan satu arah, sehingga memungkinkan ada data yang NULL (kosong) di satu sisi.

Outer Join terbagi menjadi 2 yaitu :

1. Left Join

Bentuk Umum: SELECT tabel1.*, tabel2.* FROM tabel1 LEFT JOIN tabel2 ON tabel1.PK=tabel2.FK;

HAL: 177

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN

LEFT JOIN (LANJUTAN)

Perintah menggabungkan tabel pelanggan dan pesanKita akan menggabungkan tabel **pelanggan dan pesan dimana kita akan** menampilkan daftar pelanggan yang pernah melakukan pemesanan (transaksi).

SELECT pelanggan.id_pelanggan, pelanggan.nm_pelanggan, Pesan.id_pesan, pesan.tgl_pesan FROM pelanggan LEFT JOIN pesan ON pelanggan.id_pelanggan=pesan.id_pelanggan;

HAL: 178

PERINTAH MySQL LANJUTAN

LEFT JOIN (LANJUTAN)

Hasilnya:

+		+	+
id_pelanggan	nm_pelanggan	id_pesan	tgl_pesan
P0001 P0001 P0002 P0002 P0003	Achmad Solichin Achmad Solichin Budianto Budianto Hasan Amin Riyadi	5 2 3 NULL	2008-02-02 2007-12-14 2008-02-05 2008-02-10 NULL 2008-01-20

HAL: 179

STMIK ATMA LUHUR PANGKALPINANG

Berbeda dengan hasil sebelumnya (inner join), penggunaan left join akan menampilkan juga data pelanggan dengan id P0003, walaupun pelanggan tersebut belum pernah bertransaksi. Dan pada kolom id_pesan dan tgl_pesan untuk pelanggan P0003 isinya NULL, artinya di tabel kanan (pesan) pelanggan tersebut tidak ada.

HAL: 180

PERINTAH MySQL LANJUTAN PERINTAH SELECT DENGAN JOIN (LANJUTAN)

2. Right Join

Bentuk Umum: SELECT tabel1.*, tabel2.* FROM tabel1 RIGHT JOIN tabel2 ON tabel1.PK=tabel2.FK;

HAL: 181

STMIK ATMA LUHUR PANGKALPINANG

PERINTAH MySQL LANJUTAN RIGHT JOIN (LANJUTAN)

Perintah menggabungkan tabel pelanggan dan pesan

SELECT pelanggan.id_pelanggan, pelanggan.nm_pelangga, Pesan.id_pesan, pesan.tgl_pesan FROM pelanggan RIGHT JOIN pesan ON pelanggan.id_pelanggan=pesan.id_pelanggan;

HAL: 182

PERINTAH MySQL LANJUTAN RIGHT JOIN (LANJUTAN)

Hasilnya:

+			
id pelanggan	nm pelanggan	id pesan	tgl pesan
P0002 P0002 P0004	Achmad Solichin Budianto Budianto Amin Riyadi Achmad Solichin	4	2008-02-02 2008-02-05 2008-02-10 2008-01-20 2007-12-14

HAL: 183

STMIK ATMA LUHUR PANGKALPINANG

Dengan right join, tabel yang menjadi acuan adalah tabel sebelah kanan (tabel pesan), jadi semua isi tabel pesan akan ditampilkan. Jika data pelanggan tidak ada di tabel pelanggan, maka isi tabel pesan tetap ditampilkan.

HAL: 184



PERANCANGAN BASIS DATA

KREDIT: 3 SKS

SYARAT : PENGANTAR BASIS DATA

DATA WAREHOUSE & DATA MINING

HAL: 185

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

- DATA WAREHOUSE ADALAH KUMPULAN DARI KOMPONEN-KOMPONEN PERANGKAT KERAS DAN PERANGKAT LUNAK YANG DAPAT DIGUNAKAN UNTUK MENDAPATKAN ANALISA YANG LEBIH BAIK DARI DATA YANG BERJUMLAH SANGAT BESAR SEHINGGA DAPAT MEMBUAT KEPUTUSAN YANG BAIK.
- Data Warehouse adalah relational database yang menyimpan data sekarang dan masa lalu yang berasal dari sistem oprasional(internal) dan sumber yang lain(sumber eksternal) yang didesain untuk proses Query dan analisa dan pelaporan manajemen dalam rangka pengambilan keputusan.

HAL: 186

DATA WAREHOUSE

MANFAAT DATA WAREHOUSE

- MEMAHAMI TREND BISNIS DAN MEMBUAT PERKIRAAN KEPUTUSAN YANG LEBIH BAIK
- MENGANALISA INFORMASI MENGENAI PENJUALAN HARIAN DAN MEMBUAT KEPUTUSAN YANG CEPAT DALAM MEMPENGARUHI PERFORMANCE PERUSAHAAN.

HAL: 187

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

• ISTILAH YANG BERHUBUNGAN DENGAN DATA WAREHOUSE

- DATA MART ADALAH SUATU BAGIAN PADA DATA
 WAREHOUSE YANG MENDUKUNG PEMBUATAN LAPORAN DAN
 ANALISA DATA PADA SUATU UNIT, BAGIAN, ATAU OPERASI
 PADA SUATU PERUSAHAAN.
- ON-LINE ANALYTICAL PROCESSING (OLAP) MERUPAKAN SUATU PEMROSESAN DATABASE YANG MENGGUNAKAN TABEL FAKTA DAN DIMENSI UNTUK DAPAT MENAMPILKAN BERBAGAI MACAM BENTUJ LAPORAN, ANALISIS, QUERY DARI DATA YANG BERUKURAN BESAR.
- ON LINE TRANSACTION PROCESSING (OLTP)
 MERUPAKAN SUATU PEMROSESAN YANG MENYIMPAN DATA
 MENGENAI KEGIATAN OPERSIONAL TRANSAKSI SEHARI-HARI

HAL: 188

DATA WAREHOUSE

- ISTILAH YANG BERHUBUNGAN DENGAN DATA WAREHOUSE
 - DIMENSION TABLE ADALAH TABEL YANG BERISIKAN KATEGORI DENGAN RINGKASAN DATA DETAIL YANG DAPAT DILAPORKAN.
 - FACT TABLE MERUPAKAN TABEL YANG UMUMNYA MENGANDUNG ANGKA DAN DATA HISTORY DIMANA KEY (KUNCI) YANG DIHASILKAN SANGAT UNIK, KARENA KEY TERSEBUT TERDIRI DARI FOREIGN KEY YANG MERUPAKAN PRIMARY KEY DARI BEBERAPA DIMENSION TABLE YANG BERHUBUNGAN.
 - DSS MERUPAKAN SISTEM YANG MENYEDIAKAN INFORMASI KEPADA PENGGUNA YANG MENJELASKAN BAGAIMANA SISTEM INI DAPAT MENGANALISA SITUASI DAN MENDUKUNG KEPUTUSAN YANG BAIK.

HAL: 189

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

- KARAKTERISTIK DATA WAREHOUSE
 - SUBJECT ORIENTIED (BERORIENTASI SUBJEK)

 DATA WARHOUSE BERORIENTASI SUJECT ARTINYA DATA
 WAREHOUSE DIDESAIN UNTUK MENGANALISA DATA
 BERDASARKAN SUBJECT-SUBJECT TERTENTU DALAM
 ORGANISASI, BUKAN PADA PROSES ATAU FUNGSI APLIKASI
 TERTENTU.
 - INTEGRATED (TERINTEGRASI)

 DATA WAREHOUSE DAPAT MENYIMPAN DATA-DATA YANG
 BERASAL DARI SUMBER-SUMBER YANG TERPISAH KEDALAM
 SUATU FORMAT YANG KONSISTEN DAN SALING TERINTEGRASI
 SARU DENGAN LAINNYA.

HAL: 190

DATA WAREHOUSE

KARAKTERISTIK DATA WAREHOUSE

- TIME-VARIANT (RENTANG WAKTU)
SELURUH DATA PADA DATA WAREHOUSE DAPAT DIKATAKAN
AKURAT ATAU VALID PADA RENTANG WAKTU TERTENTU.

NON-VOLATILE

NON VOLATILE MAKSUDNYA DATA PADA WAREHOUSE TIDAK DIUPDATE SECARA REAL TIME TETAPI REFRESH DARI SISTEM OPERASINAL SECARA REGULER

HAL: 191

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

- TUGAS-TUGAS DATA WAREHOUSE
 - PEMBUATAN LAPORAN
 - ON-LINE ANALYTICAL PROCESSING (OLAP)
 - DATA MINING
 - PROSES INFORMASI EXECUTIVE

HAL: 192

DATA WAREHOUSE

KEUNTUNGAN DATA WAREHOUSE

- DATA DIORGANISIR DENGAN BAIK UNTUK QUERY ANALISIS DAN SEBAGAI BAHAN UNTUK PEMROSESAN TRANSAKSI.
- PERBEDAAN <mark>DIANTARA STRUKTUR DATA Y</mark>ANG HETEROGEN PADA BEBERAPA SUMBER YANG TERPISAH DAPAT DIATASI.
- ATURAN UNTUK TRANSFORMASI DATA DITERAPKAN UNTUK MEMVALIDASI DAN MENGKONSOLIDASI DATA APABILA DATA DIPINDAHKAN DARI DATABASE OLTP KE DATA WAREHOUSE
- MASALAH KEAMANAN DAN KINERJA BISA DIPECAHKAN TANPA PERLU MENGUBAH SISTEM PRODUKSI.

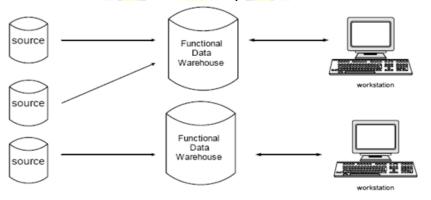
HAL: 193

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

TIGA JENIS DASAR SISTEM DATA WAREHOUSE

- FUNCTIONAL DATA WAREHOUSE (DATA WAREHOUSE

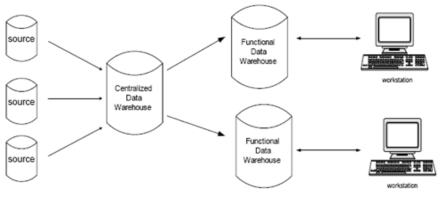


HAL: 194

DATA WAREHOUSE

TIGA JENIS DASAR SISTEM DATA WAREHOUSE

- CENTRALIZED DATA WAREHOUSE (DATA WAREHOUSE



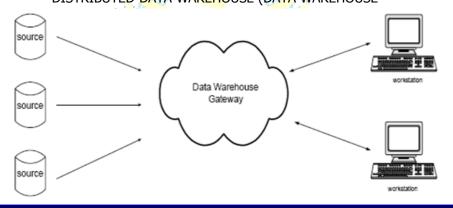
HAL: 195

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

TIGA JENIS DASAR SISTEM DATA WAREHOUSE

- DISTRIBUTED DATA WAREHOUSE (DATA WAREHOUSE



HAL: 196

DATA WAREHOUSE

METODOLOGI PERANCANGAN DATABASE

- PEMILIHAN PROSES
- PEMILIHAN SUMBER
- MENGIDENTIFIKASI DIMENSI
- PEMILIHAN FAKTA
- MENYIMPAN PRE-KALKULASI DI TABEL FAKTA
- MELENGKAPI TABEL DIMENSI
- PEMILIHAN DURASI DATABASE
- MENELUSURI PERUBAHAN DIMENSI YANG PERLAHAN
- MENENTUKAN PRIOROTAS DAN MODE QUERY

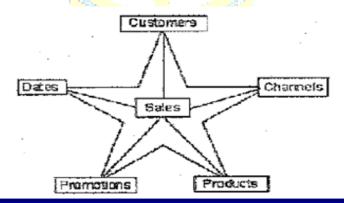
HAL: 197

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

MODEL UNTUK DATA WAREHOUSE

MODEL DIMENSIONAL

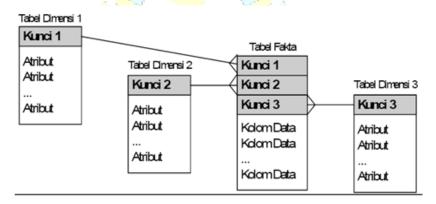


HAL: 198

DATA WAREHOUSE

MODEL UNTUK DATA WAREHOUSE

- SCHEMA BINTANG
 - SCHEMA BINTANG SEDERHANA



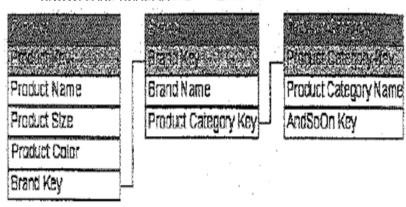
HAL: 199

STMIK ATMA LUHUR PANGKALPINANG

DATA WAREHOUSE

MODEL UNTUK DATA WAREHOUSE

SNOWFLAKE SCHEMA



HAL: 200

DATA MINING

 DATA MINING ADALAH EKSTRAKSI INFORMASI ATAU POLA YANG PENTING ATAU MENARIK DARI DATA YANG ADA DI DATABASE YANG BESAR.

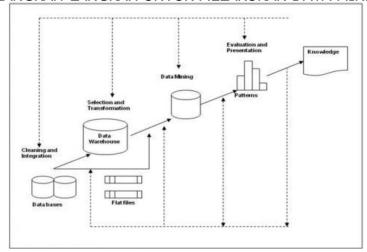


HAL: 201

STMIK ATMA LUHUR PANGKALPINANG

DATA MINING

LANGKAH-LANGKAH UNTUK MELAKUKAN DATA MINING



HAL: 202

DATA MINING

- LANGKAH-LANGKAH UNTUK MELAKUKAN DATA MINING
 - **DATA CLEANING** (UNTUK MENGHILANGKAN NOISE DATA YANG TIDAK KONSISTEN)
 - **DATA INTEGRATION** (DIMANA SUMBER DATA YANG TERPECAH DAPAT DISATUKAN)
 - DATA SELECTION (DI MANA DATA YANG RELEVAN DENGAN TUGAS ANALISIS DIKEMBALIKAN KE DALAM DATABASE)
 - DATA TRANSFORMATION (DIMANA DATA BERUBAH ATAU BERSATU MENJADI BENTUK YANG TEPAT UNTUK MENAMBANG DENGAN RINGKASAN PERFORMA ATAU OPERASI AGRESI)
 - DATA MINING (PROSES ESENSIAL DIMANA METODE YANG INTELIJEN DIGUNAKAN UNTUK MENGEKSTRAK POLA DATA)
 - PATTERN EVOLUTION (UNTUK MENGIDENTIFIKASI POLA YANG BENAR-BENAR MENARIK YANG MEWAKILI PENGETAHUAN BERDASARKAN ATAS BEBERAPA TINDAKAN YANG MENARIK)
 - KNOWLEDE PRESENTATION (DIMANA GAMBARAN TEKNIK VISUALISASI DAN PENGATAHUAN DIGUNAKAN UNTUK MEMBERIKAN PENGETAHUAN YANG TELAH DITAMBANG KEPADA USER)

HAL: 203

STMIK ATMA LUHUR PANGKALPINANG

DATA MINING

- ARSITEKTUR DATA MINING
 - DATABASE, DATA WAREHOUSE ATAU TEMPAT PENYIMPANAN INFORMASI LAINNYA.
 - SERVER DATABASE ATAU DATA WAREHOUSE
 - KNOWLEDGE BASE
 - DATA MINING ENGINE
 - PATTERN EVOLUTION MODULE
 - GRAPHICAL USER INTERFACE

HAL: 204

DATA MINING

- JENIS DATA DALAM DATA MINING
 - RELATION DATABASE: SEBUAH DATABASE, ATAU DISEBUT JUGA DATABASE MANAGEMENT SYSTEM (DBMS), MENGANDUNG SEKUMPULAN DATA YANG SALING BERHUBUNGAN DIKENAL SEBAGAI DATABASE DAN SATU SET PROGRAM PERANGKAT LUNAK UNTUK MENGATUR DAN MENGAKSES DATA TERSEBUT.
 - DATA WAREHOUSE: SEBUAH DATA WAREHOUSE MERUPAKAN SEBUAH RUANG PENYIMPANAN INFORMASI YANG TERKUMPUL DARI BERANEKA MACAM SUMBER DISIMPAN DALAM SKEMA YANG MENYATU DAN BIASANYA TERLETAK PADA SEBUAH SITE.

HAL: 205

STMIK ATMA LUHUR PANGKALPINANG

DATA MINING

- SECARA UMUM TUGAS DATA MINING DAPAT DIKLASIFIKASIKAN KE DALAM DUA KATAGORI :
 - DESKRIPTIF

ADALAH UNTUK MENGKLASIFIKASIKAN SIFAT UMUM SUATU DATA DI DALAM DATABASE.

- PREDIKTIF

ADALAH UNTUK MENGAMBIL KESIMPULAN TERHADAP DATA TERAKHIR UNTUK MEMBUAT PREDIKSI.

HAL: 206

DATA MINING

POLA YANG DAPAT DITAMBANG

- KONSEP / CLASS DESCRIPTION
- ASSOCIATION ANALYSIS
- KLASIFIKASI DAN PREDIKSI
- CLUSTER ANALYSIS
- OUTLIER ANALYSIS
- EVOLUTION ANALYSIS

HAL: 207