





Model Basis Data Relasional

Pandangan Logika Data

-  Struktur model basis data relasional dan kebebasan data memungkinkan kita untuk memandang data secara *logika* daripada secara *fisik*.
-  Pandangan secara logika membuat konsep file tentang penyimpanan data menjadi lebih sederhana.
-  Penggunaan tabel independen secara logik lebih mudah dipahami.
-  Kesederhanaan logika menghasilkan metodologi perancangan basis data lebih sederhana dan lebih efektif.

Pandangan Logika Data



Entitas dan Atribut

- ◆ **Sebuah entitas dapat berupa orang, tempat, kejadian atau sesuatu yang kita gunakan dalam mengumpulkan data.**
 - Universitas - mahasiswa, fakultas, jurusan
 - Kantor - pegawai, meja, kursi, aktifitas

- ◆ **Setiap entitas memiliki sifat-sifat khusus yang disebut atribut**
 - Mahasiswa - NIM, nama, tgl. Lahir, alamat, wali, IPK.
 - Mobil - Merek, jenis, bahan bakar, tahun pembuatan, harga.

Pandangan Logika Data



Entitas dan Atribut

- ◆ Sebuah group entitas yang saling berhubungan menjadi sebuah himpunan entitas.
 - Himpunan entitas **MAHASISWA** berisi seluruh entitas mahasiswa.
 - Himpunan entitas **FAKULTAS** berisi seluruh entitas fakultas.
 - Himpunan entitas **PEGAWAI** berisi seluruh entitas pegawai.

Pandangan Logika Data



Entitas dan Atribut

- ◆ **Tabel dan Sifat-sifatnya.**
 - **Sebuah tabel berisi sebuah group entitas yang saling berhubungan.**
 - **Istilah himpunan entitas dan tabel memiliki arti yang hampir sama dan sering diguna-kan secara bergantian.**
 - **Sebuah tabel disebut juga relasi.**

Sifat-sifat Tabel Relasional




- ❖ Sebuah tabel seolah-olah merupakan struktur dua dimensi yang terdiri dari baris dan kolom
- ❖ Setiap baris (tupel/record) mewakili sebuah entitas
- ❖ Setiap kolom mewakili atribut dan memiliki nama yang jelas
- ❖ Setiap perpotongan baris dan kolom mewakili nilai sebuah data
- ❖ Setiap tabel harus memiliki *Primary Key* yang dikenali secara unik untuk setiap baris
- ❖ Semua nilai pada kolom harus sesuai dengan format data
- ❖ Setiap kolom memiliki jangkauan nilai tertentu yang dikenal sebagai domain atribut
- ❖ Setiap baris membawa informasi yang menggambarkan satu kejadian entitas
- ❖ Urutan baris dan kolom bukan hal yang penting untuk DBMS

Contoh Nilai Atribut Tabel Mahasiswa

	Nomhs	Nama	Alamat	Sks	IPK	Kode_wali
▶	151000500	Prima Santoro	Jl.Menu 12 Demangan Yk.	20	2.51	2
	151000501	Priyani	Jl.Demangan Kidul 221 Yk.	18	2.92	2
	151000502	Lufti Yudhianto	Bantengan Lor 34 Yk.	18	3.05	3
	151000503	Sudarno Prawiro	Sagan 76 Yk.	22	2.21	3
	151000504	Yanti Komariah	Jl.Inpres 88 CT XI Yk.	21	3.42	2
	151000505	Sambudi Ratno	Jl.Perkutut 332 Yk.	18	2.02	3

Gambar 2.1. Nilai atribut tabel mahasiswa

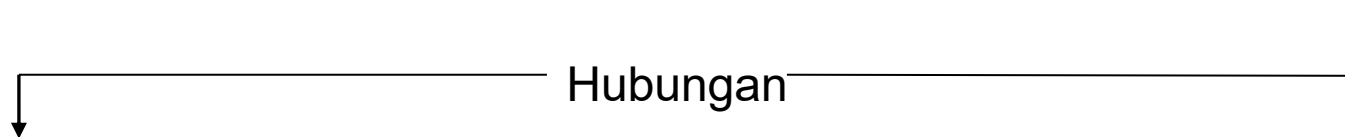
Kunci (*key*)

-  Redudansi yang terkontrol (penggunaan atribut secara bersama-sama) membuat basis data relasional berjalan dengan baik.
-  **Primary key** suatu tabel akan muncul lagi pada tabel lainnya sebagai penghubung (**foreign key**)
-  Jika **foreign key** mengandung suatu nilai atau nilai kosong yang cocok, maka tabel yang menggunakan **foreign key** tersebut dikatakan menunjukkan integritas refe-rensial.

Contoh Sederhana Basis Data Relasional

Nama tabel **SISWA**
Primary key **NOMHS**
Foreign key **KODE_WALI**

	Nomhs	Nama	Alamat	Sks	IPK	Kode_wali
▶	151000500	Prima Santoro	Jl.Menu 12 Demangan Yk.	20	2.51	2
	151000501	Priyani	Jl.Demangan Kidul 221 Yk.	18	2.92	2
	151000502	Lufti Yudhianto	Bantengan Lor 34 Yk.	18	3.05	3
	151000503	Sudarno Prawiro	Sagan 76 Yk.	22	2.21	3
	151000504	Yanti Komariah	Jl.Inpres 88 CT XI Yk.	21	3.42	2
	151000505	Sambudi Ratno	Jl.Perkutut 332 Yk.	18	2.02	3

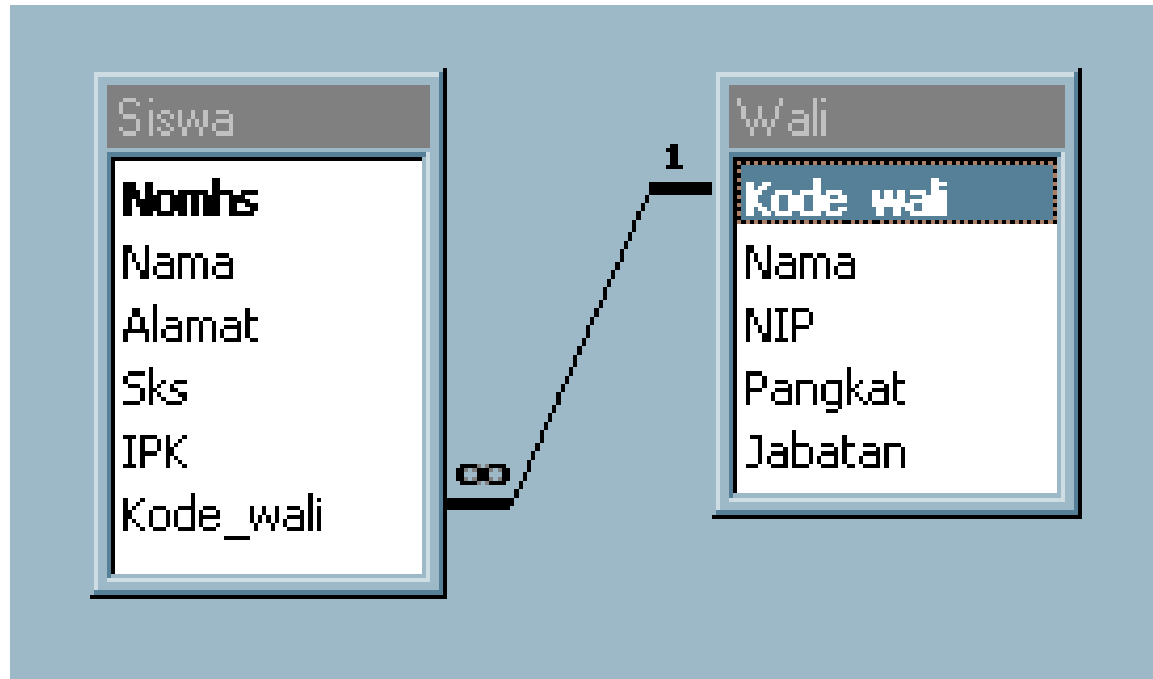


	Kode_wali	Nama	NIP	Pangkat	Jabatan
▶	1	DR. Samtidar	093065531	Lektor	Dosen
	2	Drs.Panca Wahono	094067432	Lektor Kepala	Dosen
	3	Dra.Siti Munawati	094071303	Lektor	Sesjur

Nama tabel **WALI**
Primary key **KODE_WALI**
Tidak ada foreign key




Gambar 2.2. Contoh Sederhana Basis Data Relasional

Skema Basis Data Relasional : PERWALIAN



Gambar 2.3. Skema basis data relasional : Perwalian

Kunci (*key*)

-  Sebuah kunci (*key*) dapat membantu mendefinisikan relasi entitas.
-  Aturan kunci didasarkan pada konsep yang dikenal dengan istilah determinasi, yang digunakan untuk mendefinisikan ketergantungan fungsional.
-  Ketergantungan Fungsional.
 - ◆ Atribut B bergantung secara fungsional pada A, Jika A menentukan B.
 - ◆ Sebuah atribut bagian dari suatu kunci dikenal sebagai atribut kunci.
 - ◆ Kunci yang memiliki lebih dari satu atribut disebut dengan kunci komposit.
 - ◆ Jika atribut (B) bergantung secara fungsional pada kunci komposit (A) tetapi tidak terdapat pada salah satu kunci komposit tersebut, atribut (B) bergantung penuh secara fungsional pada (A).

Kunci Basis Data Relasional

❑ *Super key*

Sebuah atribut (atau kombinasi atribut) secara unik mengenali setiap entitas dalam sebuah tabel.

❑ *Candidate key*

Sebuah *super key* minimal, yaitu *super key* yang tidak merupakan bagian atribut dari suatu *superkey*.

❑ *Primary key*

Candidate key yang terpilih untuk mengenali secara unik seluruh nilai atribut pada sebuah baris dan tidak boleh kosong

❑ *Secondary key*

Sebuah atribut (atau kombinasi atribut) secara paksa digunakan untuk tujuan pengambilan data.

❑ *Foreign key*

Sebuah atribut (atau kombinasi atribut) dalam sebuah tabel dimana nilainya cocok dengan *primary key* pada tabel lainnya

Aturan Integritas

Integritas Entitas

Syarat Seluruh entitas bersifat unik dan tidak ada primary key yang bernilai kosong.

Tujuan Menjamin setiap entitas memiliki identitas secara unik dan meyakinkan bahwa nilai setiap foreign key pasti mereferensi nilai primary key secara tepat.

Integritas Referensial

Syarat Foreign key dapat bernilai kosong sejauh bukan merupakan bagian dari primary key tabel atau memiliki nilai yang sesuai dengan nilai primary key dari suatu tabel yang terhubung.

Tujuan Membuat nilai suatu atribut yang berhubungan tidak memiliki nilai yang salah.

Ilustrasi Aturan Integritas

Table name: CUSTOMER
Primary key: CUS_CODE
Foreign key: AGENT_CODE

Database name: CH2_INSURE_CO



	CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_RENEW_DATE	AGENT_CODE
▶	10010	Ramas	Alfred	A	Friday, March 12, 1999	502
	10011	Dunne	Leona	K	Tuesday, May 23, 2000	501
	10012	Smith	Kathy	W	Tuesday, January 05, 1999	502
	10013	Olowski	Paul	F	Monday, September 20, 1999	
	10014	Orlando	Myron		Monday, December 04, 2000	501
	10015	O'Brian	Amy	B	Tuesday, August 29, 2000	503
	10016	Brown	James	G	Wednesday, March 01, 2000	502
	10017	Williams	George		Friday, June 23, 2000	503
	10018	Farriss	Anne	G	Tuesday, November 09, 1999	501
	10019	Smith	Olette	K	Friday, February 18, 2000	503

Table name: AGENT
Primary key: AGENT_CODE
Foreign key: none

	AGENT_CODE	AGENT_AREACODE	AGENT_PHONE	AGENT_LNAME	AGENT_YTD_SLS
▶	501	713	228-1249	Alby	\$1,735.00
	502	615	882-1244	Hahn	\$4,967.00
	503	615	123-5589	Okon	\$3,093.00

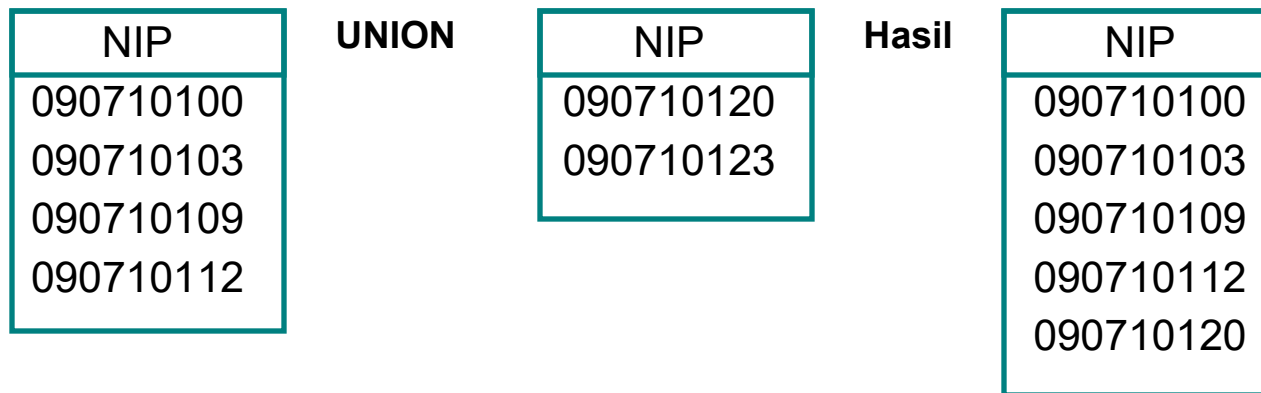
FIGURE 2.4 ■ AN ILLUSTRATION OF INTEGRITY RULES

Operator Basis Data Relasional

-  Derajat kesempurnaan relasi dapat diukur dengan seberapa luas dukungan aljabar relasionalnya.
-  Aljabar relasional mendefinisikan secara teoritis cara memanipulasi isi tabel dengan menggunakan delapan fungsi relasional: **SELECT**, **PROJECT**, **JOIN**, **INTERSECT**, **UNION**, **DIFFERENCE**, **PRODUCT** dan **DIVIDE**.

Operator Basis Data Relasional

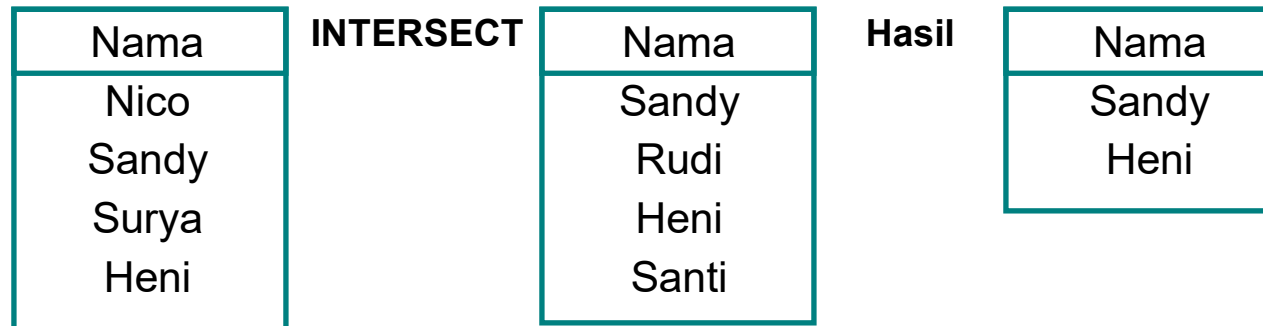
- ❑ **UNION** menggabungkan semua baris dari dua buah tabel dan kedua tabel tersebut harus sesuai.



Gambar 2.5. Union

Operator Basis Data Relasional

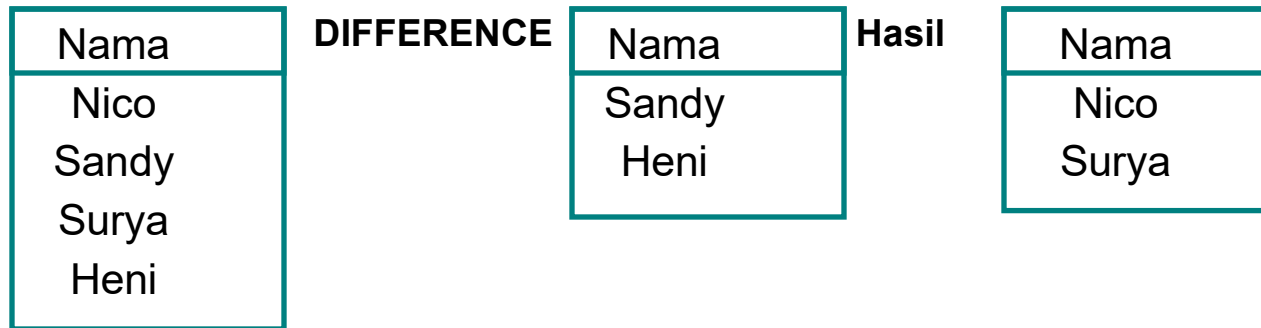
- ❑ **INTERSECT** menghasilkan sebuah daftar yang berisi hanya *record-record* yang terdapat pada kedua tabel dan kedua tabel tersebut harus sesuai.



Gambar 2.6. Intersect

Operator Basis Data Relasional

 **DIFFERENCE** menghasilkan semua record yang terdapat pada satu tabel tetapi tidak terdapat pada tabel lainnya dan kedua tabel tersebut harus sesuai.



Gambar 2.7. Difference

Operator Basis Data Relasional

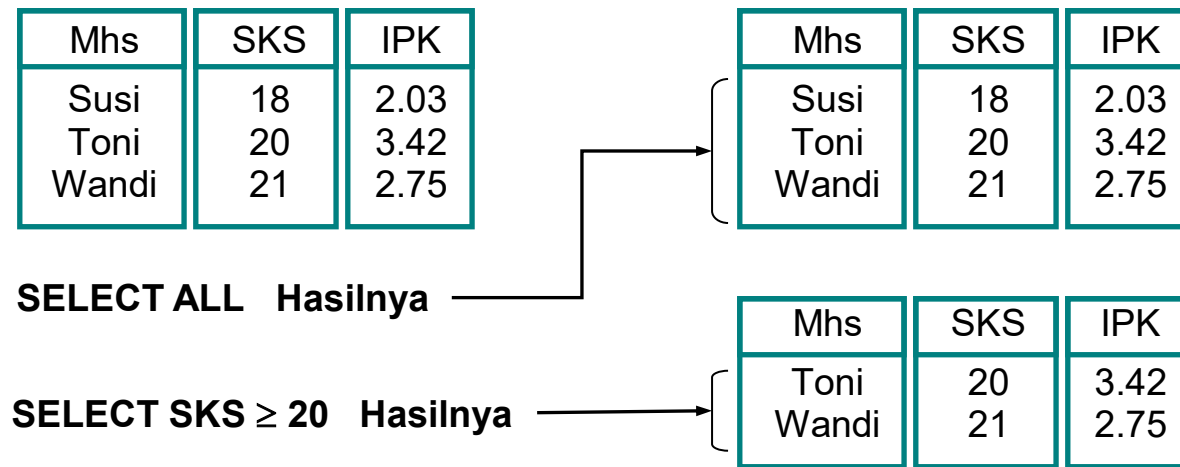
 **PRODUCT** menghasilkan sebuah daftar semua pasangan record dua buah tabel.

Wali	PRODUCT	Mhs	SKS	IPK	Hasil	Wali	Mhs	SKS	IPK
Winata		Susi	18	2.03		Winata	Susi	18	2.03
Adelia		Toni	20	3.42		Winata	Toni	20	3.42
		Wandi	21	2.75		Winata	Wandi	21	2.75
						Adelia	Susi	18	2.03
						Adelia	Toni	20	3.42
						Adelia	Wandi	21	2.75

Gambar 2.8. Product

Operator Basis Data Relasional

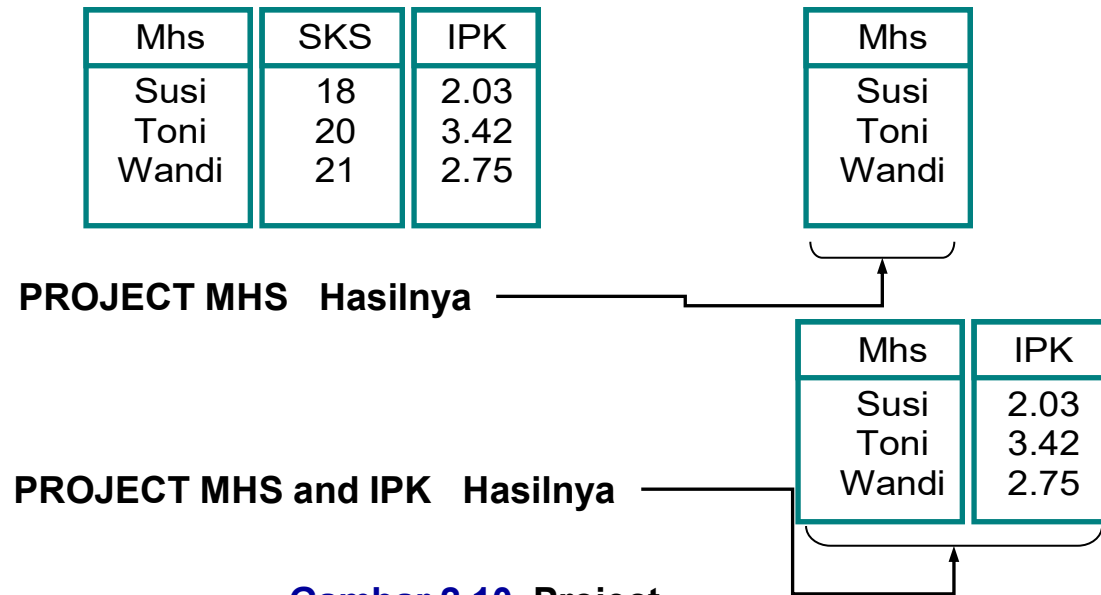
 **SELECT** menghasilkan nilai untuk semua atribut yang ditemukan dalam tabel.



Gambar 2.9. Select

Operator Basis Data Relasional

 **PROJECT** menghasilkan daftar semua nilai untuk atribut yang dipilih.



Gambar 2.10. Project

Operator Basis Data Relasional

- 🖥️ **JOIN** memungkinkan kita untuk mengkombinasikan informasi dari dua tabel atau lebih. **JOIN** memiliki kemampuan nyata untuk mendukung basis data relasional, memungkinkan penggunaan tabel independen yang dihubungkan melalui atribut yang sama.
- ◆ **Natural JOIN** menghubungkan tabel dengan memilih hanya **record** dengan nilai yang digunakan bersama-sama pada atribut yang sama. Operator ini akan menghasilkan tiga tahapan proses:
 - **PRODUCT**
 - **SELECT**
 - **PROJECT**

Operator Basis Data Relasional

 Contoh dua tabel yang akan digunakan untuk ilustrasi JOIN:

Nama Tabel : SISWA

Mhs	SKS	IPK	Kode_Wali
Wandi	21	2.75	1
Toni	20	3.42	1
Rudi	18	2.45	2
Susi	18	2.03	2
Tatik	22	3.68	4

Nama Tabel : WALI

Kode_Wali	Wali
1	Suminto
2	Farid
3	Ganjar

Gambar 2.11. Tabel untuk ilustrasi join

Operator Basis Data Relasional

Natural JOIN, Tahap 1: PRODUCT

Mhs	SKS	IPK	Kode_Wali	Kode_Wali	Wali
Wandi	21	2.75	1	1	Suminto
Toni	20	3.42	1	1	Suminto
Rudi	18	2.45	2	1	Suminto
Susi	18	2.03	2	1	Suminto
Tatik	22	3.68	4	1	Suminto
Wandi	21	2.75	1	2	Farid
Toni	20	3.42	1	2	Farid
Rudi	18	2.45	2	2	Farid
Susi	18	2.03	2	2	Farid
Tatik	22	3.68	4	2	Farid
Wandi	21	2.75	1	3	Ganjar
Toni	20	3.42	1	3	Ganjar
Rudi	18	2.45	2	3	Ganjar
Susi	18	2.03	2	3	Ganjar
Tatik	22	3.68	4	3	Ganjar

Gambar 2.12. Natural join, tahap 1: Product

Operator Basis Data Relasional

Natural JOIN, Tahap 2: SELECT

Mhs	SKS	IPK	Kode_Wali	Kode_Wali	Wali
Wandi	21	2.75	1	1	Suminto
Toni	20	3.42	1	1	Suminto
Rudi	18	2.45	2	2	Farid
Susi	18	2.03	2	2	Farid

Gambar 2.13. Natural join, tahap 2: Select

Operator Basis Data Relasional

Natural JOIN, Tahap 3: PROJECT

Mhs	SKS	IPK	Kode_Wali	Wali
Wandi	21	2.75	1	Suminto
Toni	20	3.42	1	Suminto
Rudi	18	2.45	2	Farid
Susi	18	2.03	2	Farid

Gambar 2.14. Natural join, tahap 3: Project

Operator Basis Data Relasional

- ◆ **EquiJOIN** menghubungkan tabel didasarkan pada kondisi yang sama dengan membandingkan kolom tertentu setiap tabel. Hasil **equiJOIN** tidak menghilangkan kolom duplikat dan kondisi atau kriteria penggabungan tabel harus terdefinisi secara eksplisit.
- ◆ **Theta JOIN** adalah **equiJOIN** yang membandingkan kolom tertentu setiap tabel menggunakan operator pembandingan selain operator sama dengan.
- ◆ Pada **Outer JOIN** pasangan data yang tidak cocok akan tetap dipertahankan dan nilai untuk tabel lainnya yang tidak cocok akan dibiarkan kosong.

Operator Basis Data Relasional

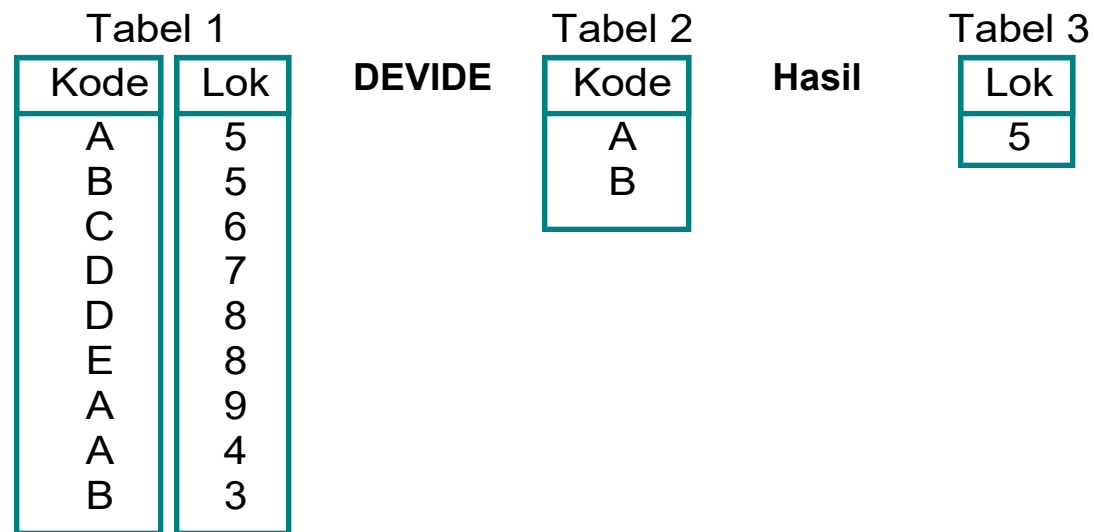
Outer JOIN

Mhs	SKS	IPK	Kode_Wali	Wali
Wandi	21	2.75	1	Suminto
Toni	20	3.42	1	Suminto
Rudi	18	2.45	2	Farid
Susi	18	2.03	2	Farid
Tatik	22	3.68	4	Ganjar

Gambar 2.15. Outer Join



Operator Basis Data Relasional

 **DIVIDE** memerlukan dua buah tabel yang masing-masing terdiri dari satu dan dua kolom. Perhatikan ilustrasi berikut ini:



Gambar 2.16. Devide

Kamus Data dan Katalog Sistem

-  **Kamus data berisi metadata untuk menjelaskan secara detail catatan semua tabel di dalam suatu basis data.**
-  **Katalog sistem adalah kamus data sistem yang sangat detail yang menggambarkan semua objek di dalam suatu basis data.**
 - ◆ Basis data yang dibentuk sistem dimana tabel-tabelnya menyimpan isi dan sifat-sifat basis data.**
 - ◆ Tabel-tabelnya dapat diproses seperti tabel-tabel lainnya.**
 - ◆ Secara otomatis menghasilkan dokumentasi basis data**

Contoh Kamus Data

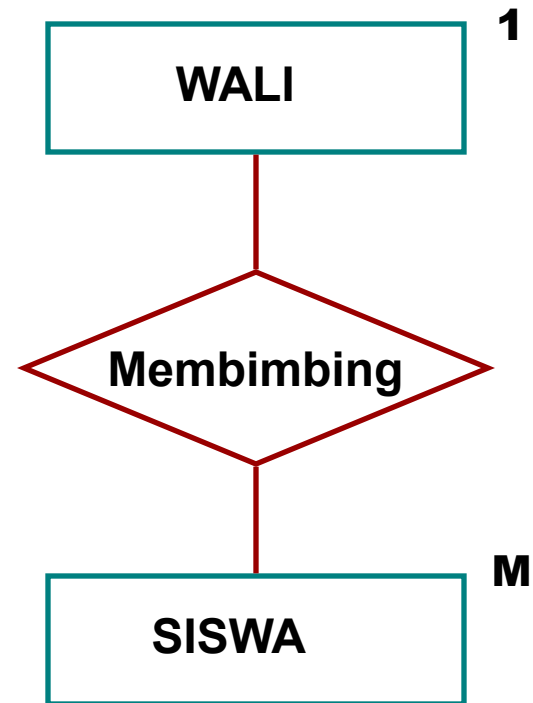
Nama Tabel	Nama Atribut	Uraian	Tipe	Format	Range	PK/ FK	Tabel Referensi
Siswa	Nomhs	Nomor mhs	Char(9)	999999999		PK	
	Nama	Nama mhs	Varchar(18)	Xxxxxxxxxx			
	SKS	Jumlah SKS	Byte	99	2-24		
	IPK	IP.Kumulatif	Number(4,2)	9.99	0.00-4.00		
	Kd_Wali	Kode wali	Number(2)	99	1-99	FK	Wali
Wali	Kd_Wali	Kode wali	Number(2)	99	1-99	PK	
	NIP	No. Pegawai	Char(9)	999999999			
	Nama	Nama wali	Varchar(18)	Xxxxxxxxxx			
	Jbt_Fng	Fungsional	Varchar(15)	Xxxxxxxxxx			

Relasi pada Basis Data Relasional

E-R Diagram (ERD)

- ◆ Bentuk persegi panjang mewakili entitas.
- ◆ Nama entitas berupa kata benda dan huruf besar.
- ◆ Bentuk berlian digunakan untuk mewakili relasi antar entitas.
- ◆ Angka 1 digunakan untuk mewakili “1” (satu) data pada relasi.
- ◆ Huruf M digunakan untuk mewakili “many” (banyak) data dari relasi.

Relasi Antara Wali dan Siswa



Gambar 2.17. Relasi antara wali dan siswa

Cara Lain Untuk Menyatakan Relasi Antara Wali dan Siswa



Gambar 2.18. Bentuk lain relasi antara wali dan siswa

Relasi 1: M : Basis Data Perwalian

Tabel : WALI
PK : KODE_WALI
FK : -

	Kode_wali	Nama	NIP	Pangkat	Jabatan
▶	1	DR. Samtidar	093065531	Lektor	Dosen
	2	Drs. Panca Wahono	094067432	Lektor Kepala	Dosen
	3	Dra. Siti Munawati	094071303	Lektor	Sesjur

Hubungan

Tabel : SISWA
PK : NOMHS
FK : KODE_WALI

	Nomhs	Nama	Alamat	Sks	IPK	Kode_wali
▶	151000500	Prima Santoro	Jl. Menu 12 Demangan Yk.	20	2.51	2
	151000501	Priyani	Jl. Demangan Kidul 221 Yk.	18	2.92	2
	151000502	Lufti Yudhianto	Bantengan Lor 34 Yk.	18	3.05	3
	151000503	Sudarno Prawiro	Sagan 76 Yk.	22	2.21	3
	151000504	Yanti Komariah	Jl. Inpres 88 CT XI Yk.	21	3.42	2
	151000505	Sambudi Ratno	Jl. Perkutut 332 Yk.	18	2.02	3

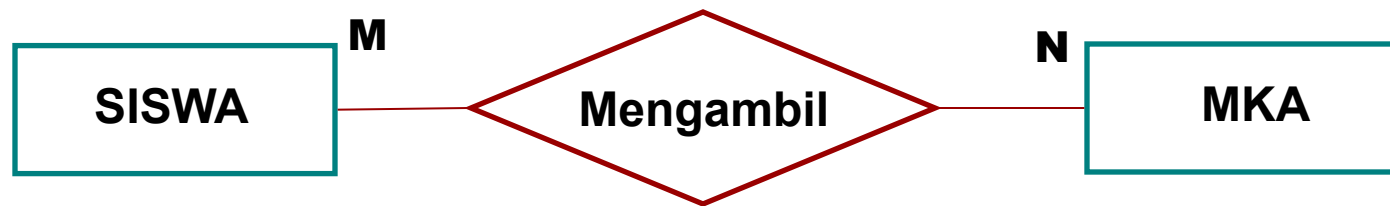
Gambar 2.19. Relasi 1: M untuk basis data perwalian

Relasi 1:M Antara Kursus dan Kelas



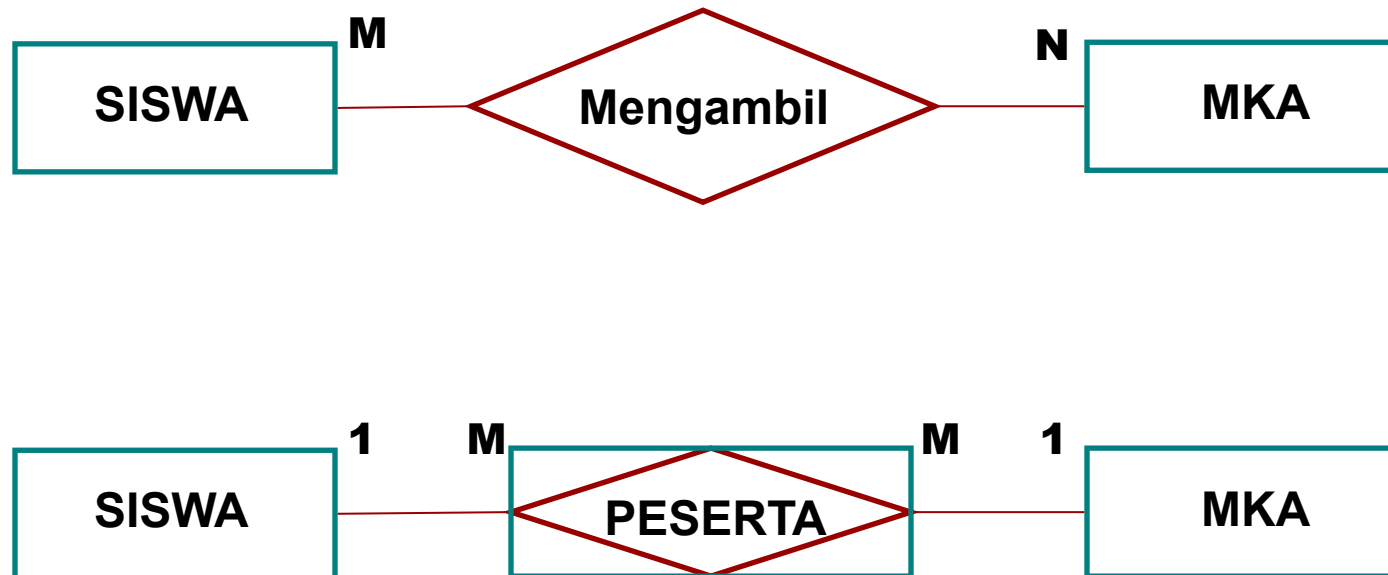
Gambar 2.20. Relasi 1:M antara kursus dan kelas

Relasi M:N Antara Siswa dan MKA



Gambar 2.21. Relasi M:N antara siswa dan MKA

Contoh Data Pengambilan MKA



Gambar 2.21. Relasi M:N antara siswa dan MKA

Relasi “Many to Many” Antara Siswa dan MKA

Tabel SISWA

	Nomhs	Nama	Kode_Mka
▶	123900010	Jamhari	2231043
	123900010	Jamhari	4232023
	123900010	Jamhari	4232043
	123900010	Jamhari	6232102
	123900015	Verawati	2231043
	123900015	Verawati	4232023
	123900015	Verawati	4232043
	123900015	Verawati	6232102

Basis Data KRP

Tabel MKA

	Kode_Mka	Mka	Sks	Dosen	Hari	Nomhs
▶	2231043	Aljabar Linier	3	DR. Singgih Wardi, MT.	Senin	123900010
	2231043	Aljabar Linier	3	DR. Singgih Wardi, MT.	Senin	123900015
	4232023	Analisa Algoritma	3	DR. M. Halid, MSc.	Kamis	123900010
	4232023	Analisa Algoritma	3	DR. M. Halid, MSc.	Kamis	123900015
	4232043	Basis Data	3	Ir. Suryana, MT.	Selasa	123900010
	4232043	Basis Data	3	Ir. Suryana, MT.	Selasa	123900015
	6232102	Pemrograman Web	2	Ir. A. Retni, MKom.	Rabu	123900010
	6232102	Pemrograman Web	2	Ir. A. Retni, MKom.	Rabu	123900015

Gambar 2.22. Relasi “many to many” antara siswa dan MKA

Konversi Relasi M:N Menjadi Dua Relasi 1:M

Nomhs	Nama	Kode_Mka
123900010	Jamhari	6232102
123900015	Verawati	4232043

Tabel SISWA
Primary key NOMHS

Kode_Mka	Nomhs
2231043	123900010
2231043	123900015
4232023	123900010
4232023	123900015
4232043	123900010
4232043	123900015
6232102	123900010
6232102	123900015

Tabel KULIAH
Primary key KODE_MKA+NOMHS
Foreign key KODE_MKA,NOMHS

Tabel MKA
Primary key KODE_MKA

Kode_Mka	Mka	Sks	Dosen	Hari
2231043	Aljabar Linier	3	DR. Singgih Wardi, MT.	Senin
4232023	Analisa Algoritma	3	DR. M. Halid, MSc.	Kamis
4232043	Basis Data	3	Ir. Suryana, MT.	Selasa
6232102	Pemrograman Web	2	Ir. A. Retni, MKom.	Rabu

Gambar 2.23. Konversi relasi M:N menjadi relasi 1:M