

A close-up photograph of a hand holding a black fountain pen with gold accents, writing on a document. The document appears to be a technical drawing or a table with some faint text and numbers. The background is blurred, showing what might be a desk or a computer monitor.

# Perancangan Perangkat Lunak



# KAMUS DATA

# Pendahuluan

- Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga user dan analis sistem mempunyai pengertian yang sama tentang input, output, dan komponen data store.



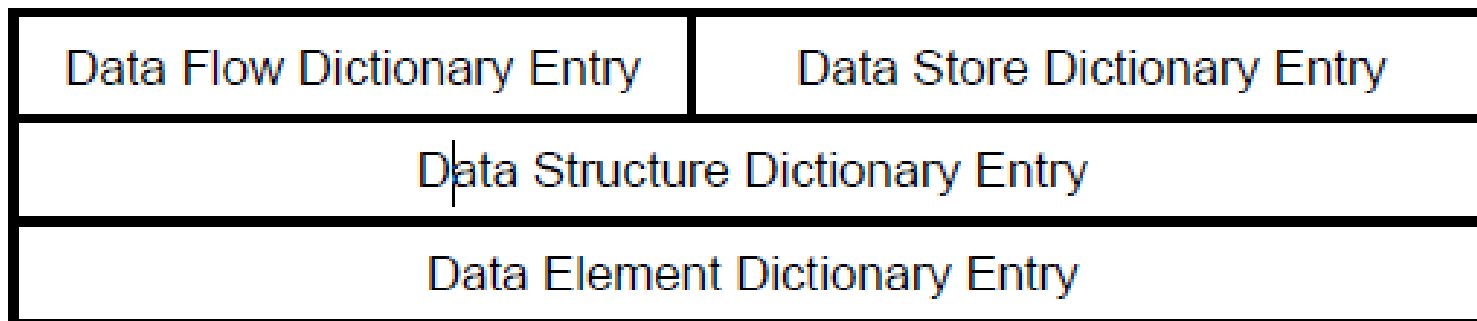
# Pendahuluan

- Pada tahap analisis, kamus data merupakan alat komunikasi antara user dan analis sistem tentang data yang mengalir di dalam sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh user. Sementara itu, pada tahap perancangan sistem kamus data digunakan untuk merancang input, laporan dan database.



# Bentuk Kamus Data

- Suatu sistem dapat diuraikan ke dalam 4 form kamus data yang menerangkan isi database sistem dalam bentuk hirarki seperti yang digambarkan sebagai berikut :



Gambar 1. Hirarki dari Form Kamus Data

# Data Flow Dictionary Entry

A close-up photograph of a person's hand holding a black pen with a gold nib, writing on a white document. The document has some faint, illegible text on it. The background is blurred, showing what appears to be a desk or table.

- Data flow dictionary entry ini menerangkan setiap data flow pada DFD. Data flow ini dapat berupa :
  - a. Satu struktur yang terdiri dari satu elemen data tunggal.
  - b. Satu struktur yang terdiri dari satu paket elemen data.
  - c. Multiple struktur.

# Data Flow Dictionary Entry

A close-up photograph of a person's hand holding a black pen with a gold nib, writing on a white document. The document has some faint, illegible text on it. The background is blurred, showing what appears to be a desk or table.

- Berdasarkan uraian di atas, maka hubungan antara alur data pada DFD dan alur data pada elemen kamus data adalah one to one relationship (relasi satu-satu). Jika DFD berisi 40 alur data, maka kamus data harus mempunyai 40 elemen alur data.

# Data Flow Dictionary Entry



- Data flow dictionary entry berisi hanya summary data atau data ringkasan, dan menerangkan alur yang mengidentifikasi dari mana alur itu berasal dan kemana alur itu menuju. Contoh sbb:

DATA FLOW DICTIONARY ENTRY	
Use	: To describe each data flow in a data flow diagram
DATA FLOW NAME	: Sales orders
DESCRIPTION	: The documents that are filled out by the customers to identify the products, and the quantities of each, that they wish to purchase.
FROM	: 1. Open mail
TO	: 2. Enter sales order data
DATA STRUCTURES	: Sales order record
COMMENTS	:



# Data Store Dictionary Entry

A close-up photograph of a person's hand holding a black pen with a gold nib, writing on a white document. The document has some faint, illegible text on it. The background is blurred, showing what appears to be a desk or table.

- Data store dictionary entry menerangkan setiap data store yang unik dalam DFD. Jika data store yang sama muncul lebih dari satu, maka hanya satu bentuk tunggal yang akan digunakan.
- Seperti halnya data flow dictionary entry, data store dictionary entry hanya berisi summary data.

# Data Store Dictionary Entry

- Contoh

## DATA STORE DICTIONARY ENTRY

Use : describe each unique data store in a data flow diagram.

DATA STORE NAME : Sales order form file  
DESCRIPTION : The history file oh sales order forms, after the data has been entered into the sales commission system.  
DATA STRUCTURES : Sales order record.  
VOLUME : Approximately 140 per day  
ACTIVITY :  
ACCESS : Order department personnel  
COMMENTS :

# Data Structure Dictionary Entry



- Data structure dictionary entry ini dilengkapi dengan setiap struktur yang ada pada bentuk data store dan data flow.
- Tujuan dari data structure dictionary entry adalah untuk menghubungkan summary description (deskripsi ringkasan) dari data flow dan data store dictionary entry ke deskripsi detail dari data element dictionary entry.

# Data Structure Dictionary Entry



- Contoh

## DATA STRUCTURE DICTIONARY ENTRY

Use : To describe each unique data structure that exists in  
(1) data flows and (2) data stores.

STRUCTURE NAME : Sales order record.  
DESCRIPTION : The sales order form that customer  
uses to order merchandise.  
DATA ELEMENTS : CUSTOMER.NUMBER  
CUSTOMER.ORDER.NUMBER  
SALESPERSON.NUMBER  
CUSTOMER.ORDER.DATE  
\*ITEM.NUMBER  
\*ITEM.DESCRPTION  
\*ITEM.QUANTITY  
\*ITEM.UNIT.PRICE  
\*ITEM.EXTENDED.PRICE  
COMMENTS : Elements marked with asterisks occur  
for each item record.

# Data Element Dictionary Entry

A close-up photograph of a person's hand holding a black pen with a gold nib, writing on a white document. The document has some faint, illegible text on it. The background is blurred, showing what appears to be a desk or table.

- Data element dictionary entry menyediakan dasar untuk skema database. Bentuk ini menyediakan data element dictionary (DED) dari kamus data yang berdasarkan komputer.

# Data Element Dictionary Entry

A close-up photograph of a person's hand holding a black pen with a gold nib, writing on a white document. The document has some faint, illegible text on it. The background is blurred, showing what appears to be a desk or table.

- Tujuan dari data element dictionary entry adalah untuk menstandarkan deskripsi dari suatu elemen sehingga elemen itu direferensikan dengan cara yang sama setiap kali digunakan.
- Hal ini sangat penting, khususnya jika suatu sistem dikembangkan dan dimaintain oleh sekelompok user dan information specialists.

# Data Element Dictionary Entry



- Contoh:

## DATA ELEMENTS DICTIONARY ENTRY

Use : To describe each unique data element contained in a data structure

DATA ELEMENT NAME	:	SALESPERSON.NUMBER
DESCRIPTION	:	The number that identifies the salesperson
TYPE	:	Numeric
LENGTH	:	4
NO. DECIMAL POS	:	
ALIASES	:	Salesman Number, Sales Rep Number
RANGE OF VALUES	:	0001-9999
TYPICAL VALUES	:	
SPECIFIC VALUES	:	
OTHER EDITING DETAILS	:	

# Pendefinisian Data Element

- Menguraikan arti dari alur data dan data store dalam DFD
- Menguraikan komposisi paket data pada alur data ke dalam alur yang lebih elementary (kecil) contoh : alamat langganan yang terdiri dari nama jalan, kota dan kode pos.
- Menguraikan komposisi paket data dalam data store.
- Menspesifikasikan nilai dan unit informasi dalam alur data dan data store.
- Menguraikan hubungan yang terinci antara data store dalam suatu entity relationship diagram (ERD)





# Notasi-Notasi Kamus Data

Notasi	Arti
=	terdiri dari, terbentuk dari, sama dengan
+	dan
()	optional
{ }	iterasi/ pengulangan, misal : 1 { ... } 10
[ ]	pilih satu dari beberapa alternatif (pilihan) misal : [AIBICID]
**	komentar
@	identifier suatu data store
	pemisah dalam bentuk [ ]
alias	nama lain untuk suatu data

# Contoh Kamus Data

- Nama = Nama\_Depan + Nama\_Belakang
- Current\_Height = \*Unit : 150 Cm\*
- Nama\_Langganan = (Title) + Nama\_Depan + (Nama\_Tengah) + Nama\_Belakang
- Customer\_Address = (Shipping\_Address) + (Billing\_Address)
- Order = Customer\_Name + Shipping\_Address + 1{item}10
- Jenis\_Kelamin = [ Pria | Wanita ]



# Contoh Kamus Data

- Penjualan = \*Jumlah penjualan selama satu tahun\*
- Pajak rate = \*Satuan pajak yang berlaku ditentukan oleh pemerintah dalam %\*
- Jumlah Pajak = \*Jumlah pajak yang harus dibayar hasil perkalian dari sales\*
- Client = Alias untuk customer.





ERD

ENTITY RELATIONSHIP  
DIAGRAM

# Data modelling vs Process Modelling

- Pemodelan proses (Co. DFD) menunjukkan simpanan data, bagaimana, dimana, dan kapan data digunakan atau diubah dalam sebuah Sistem Informasi
- Pemodelan data (Co. ER) menunjukkan definisi, struktur, dan hubungan dalam data



# Tahapan Peranc. Database (1)

## Phase 1 :

Requirements,  
Collection and Analysis

## Phase 2 :

Conceptual Database Design

## Phase 3 :

Choice of DBMS

## Phase 4 :

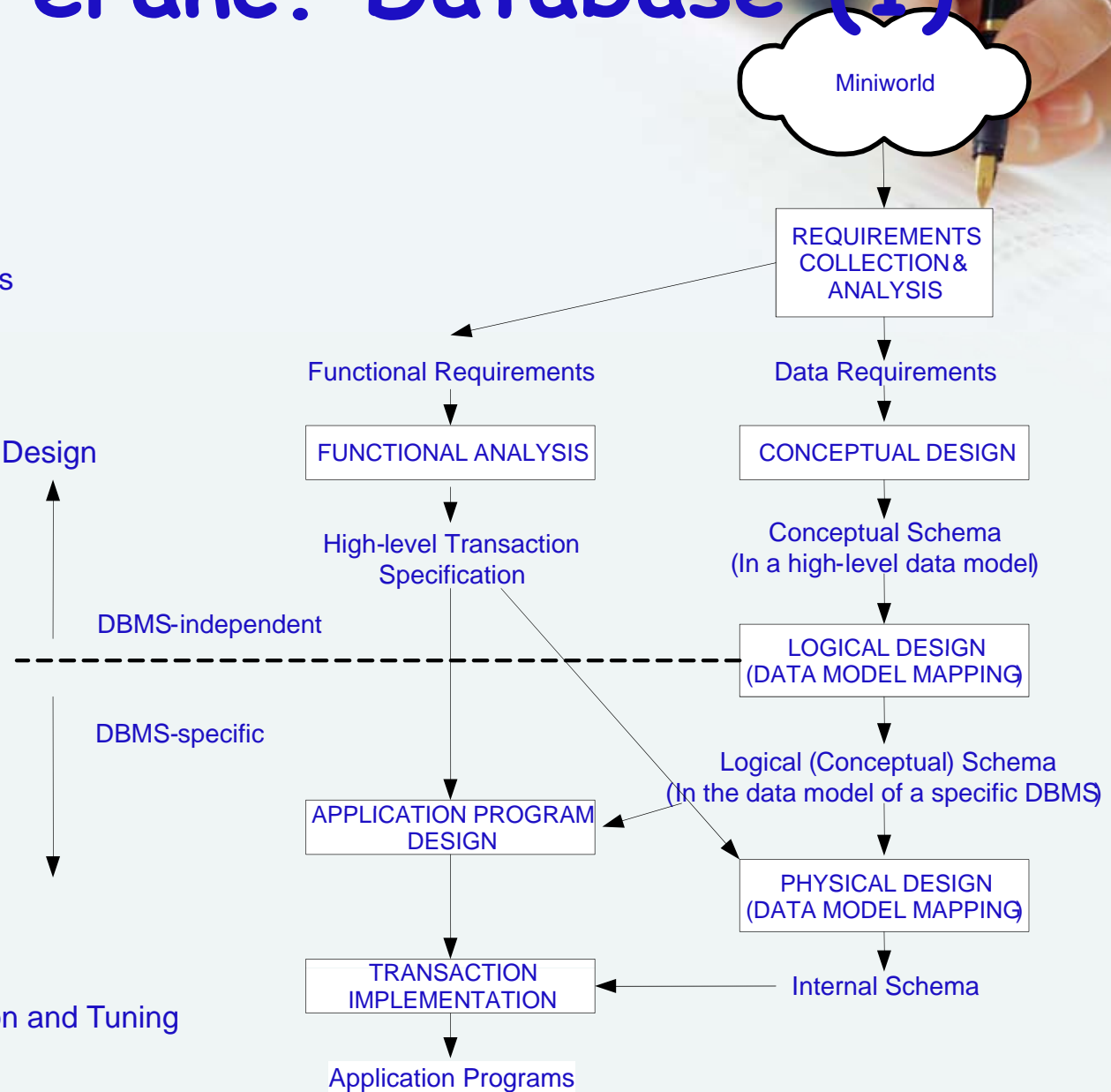
Data Model Mapping  
(Logical design)

## Phase 5 :

Physical Design

## Phase 6 :

System Implementation and Tuning



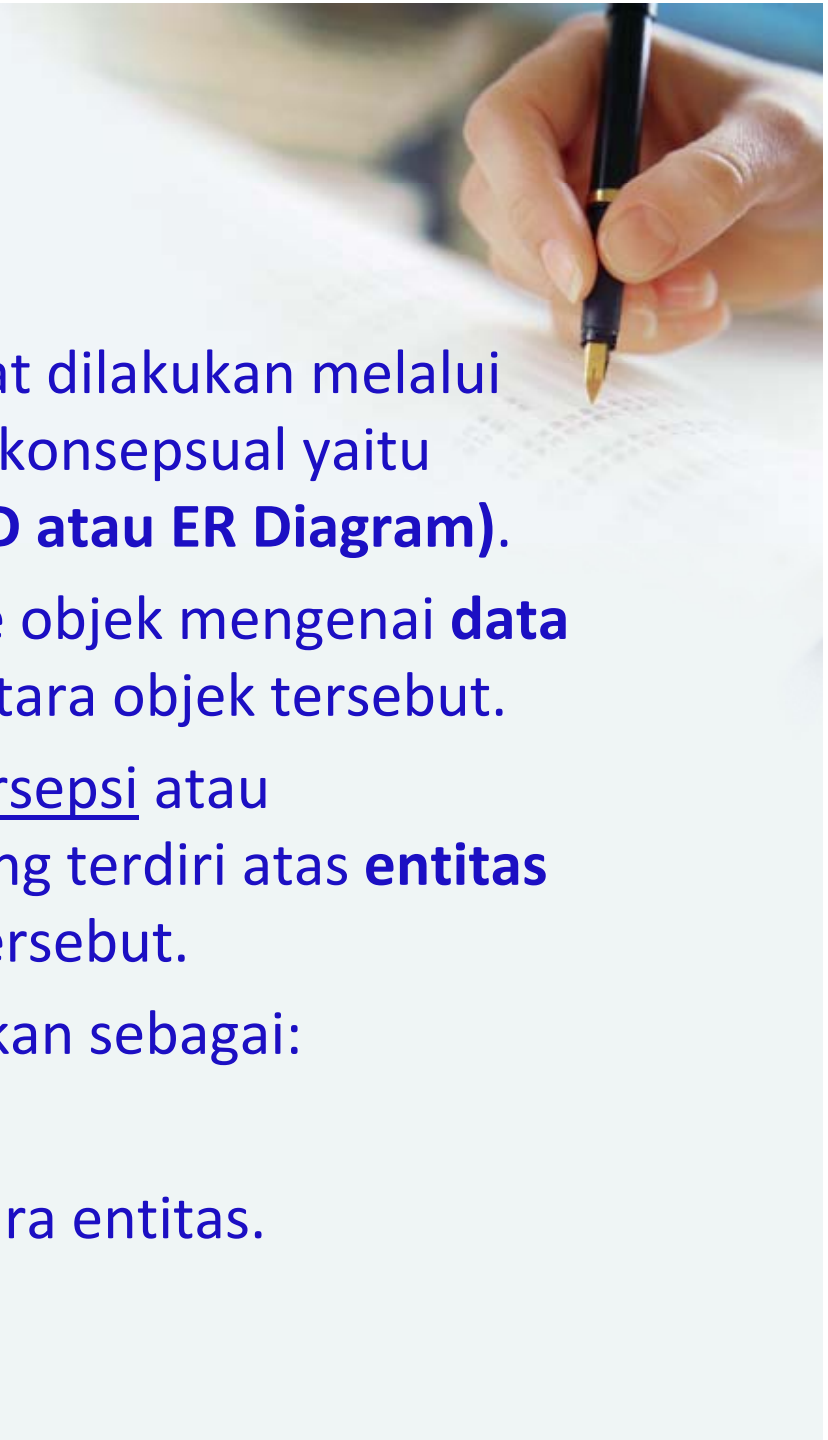
# Tahapan Peranc. Database (2)



- Hasil dari tahap requirement dan analisa berupa data-data kebutuhan user yang akan ditampung dan digambarkan pada tahap rancangan **skema konseptual (Conceptual Design)**.
- Pada tahap Conceptual Design, berisi detail deskripsi dari tipe-tipe entity, relasi dan constraint (batasan). Hasil dari tahap ini berupa rancangan skema konseptual Database (ER Diagram).
- Setelah ER Diagram yang dibuat dari rancangan skema **konseptual** database, perlu dilakukan proses **mapping** ke **skema relasi** agar database tersebut dapat diimplementasikan dengan **Relational DBMS (RDBMS)**. Tahap ini disebut Logical Design (Data Model Mapping). Hasil dari tahapan ini berupa Skema Fisik Database.
- Tahap akhir adalah Physical Design, berupa pendefinisian struktur internal storage, index, path serta organisasi file-file dalam database.

# ER Data Model

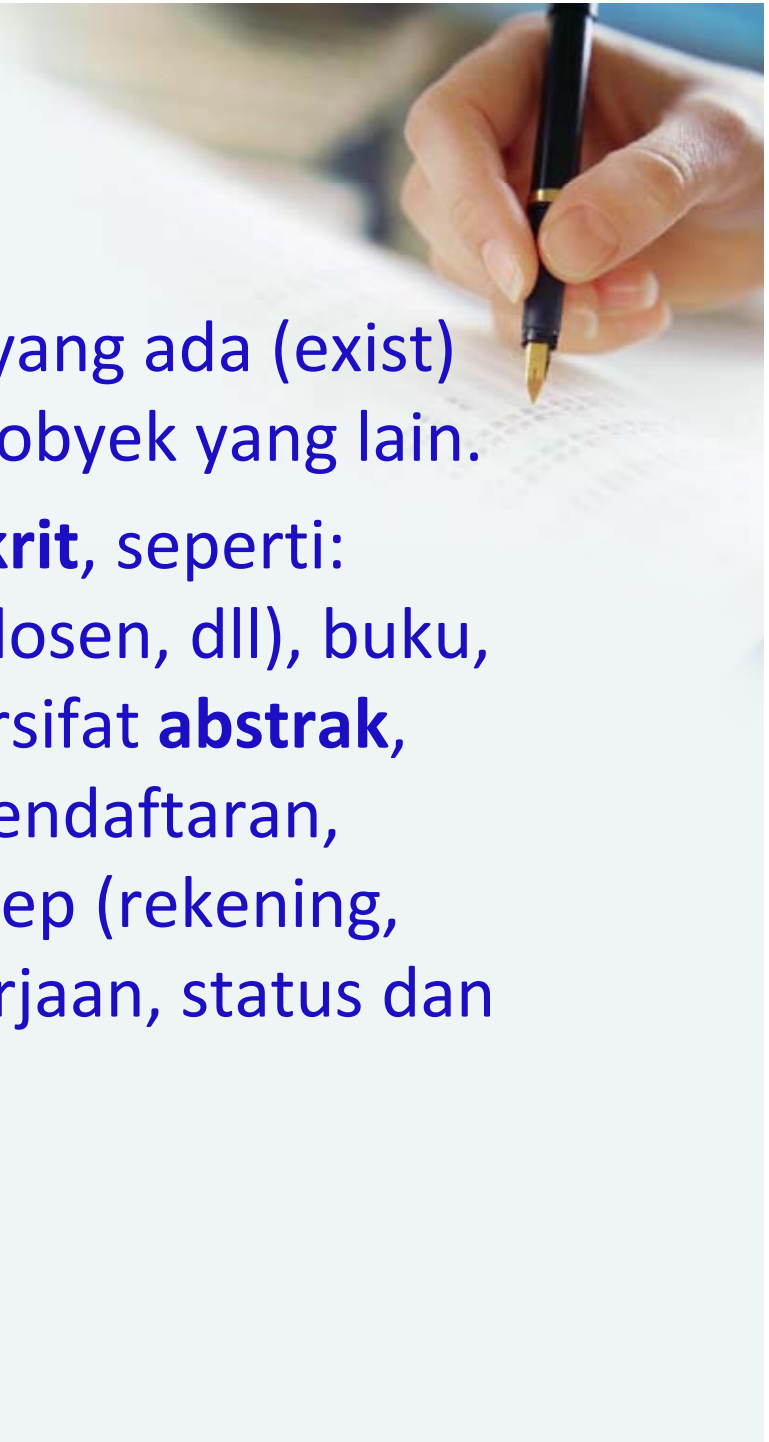
- Pemodelan sistem database dapat dilakukan melalui pendekatan perancangan secara konseptual yaitu **Entity Relationship Diagram (ERD atau ER Diagram)**.
- ER Diagram menggambarkan tipe objek mengenai **data** itu di manajemen, serta **relasi** antara objek tersebut.
- ER Model dibuat berdasarkan persepsi atau pengamatan kondisi riil/nyata yang terdiri atas **entitas** dan **relasi** antar entitas-entitas tersebut.
- Sebuah database dapat dimodelkan sebagai:
  - Kumpulan **Entity/Entitas**,
  - **Relationship/Relasi** diantara entitas.





# ER Data Model

- **Entitas** adalah sebuah obyek yang ada (exist) dan dapat dibedakan dengan obyek yang lain.
- Entitas ada yang bersifat **konkrit**, seperti: orang (pegawai, mahasiswa, dosen, dll), buku, perusahaan; dan ada yang bersifat **abstrak**, seperti: peristiwa/kejadian (pendaftaran, pemesanan, penagihan), konsep (rekening, kualifikasi), mata kuliah, pekerjaan, status dan sebagainya.



# ER Data Model

- Setiap **entitas** memiliki **atribut** sebagai keterangan dari entitas, misal. entitas mahasiswa, yang memiliki atribut: NIM, nama dan alamat.
- Setiap atribut pada entitas memiliki **kunci atribut (key atribut)** yang bersifat **unik**.



# ER Data Model

- Beberapa atribut juga dapat ditetapkan sebagai **calon kunci (candidate key)**.

Misal.

- Entitas **Mahasiswa** dengan atribut **NIM** sebagai key atribut
- Entitas **Dosen** dengan **NIP** sebagai key atribut, dan sebagainya.
- Beberapa entitas kemungkinan tidak memiliki atribut kunci sendiri, entitas demikian disebut **Entitas Lemah (Weak Entity)**.



# Symbol ER Data Model

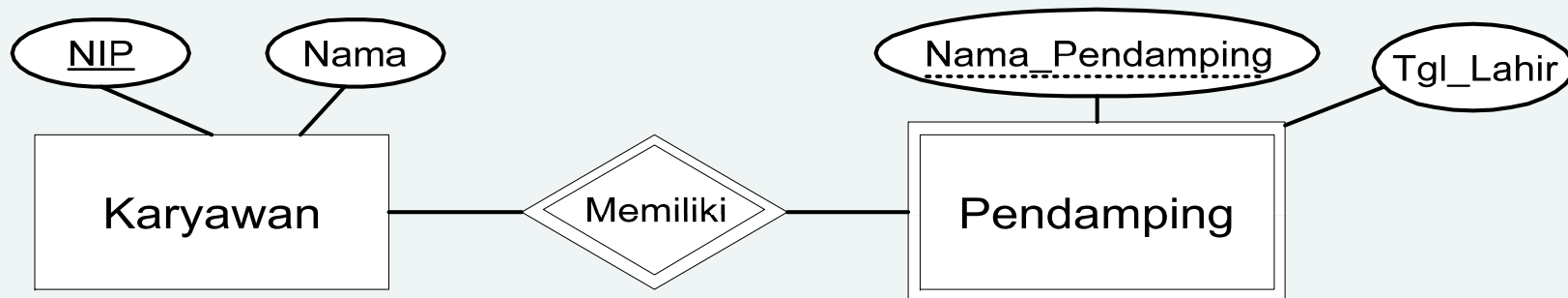


Symbol	Meaning
	ENTITY TYPE
	WEAK ENTITY TYPE
	RELATIONSHIP TYPE
	IDENTIFYING RELATIONSHIP TYPE
	ATTRIBUTE
	KEY ATTRIBUTE
	MULTIVALUED ATTRIBUTE
	COMPOSITE ATTRIBUTE
	DERIVED ATTRIBUTE
	TOTAL PARTICIPATION OF E <sub>2</sub> IN R
	CARDINALITY RATIO 1:N FOR E <sub>1</sub> :E <sub>2</sub> IN R
	STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R

# Entitas Lemah (Weak Entity)

- **Entitas Lemah (Weak Entity)** adalah entitas yang keberadaannya sangat bergantung dengan entitas lain.
  - Tidak memiliki **Key Attribute** sendiri.
  - Entitas tempat bergantung disebut **Identifying Owner/Owner**.
  - Entitas lemah tidak memiliki identifier-nya sendiri.
  - Atribut entitas lemah berperan sebagai **Partial Identifier** (identifier yang berfungsi secara sebagian).

Contoh:



# Jenis-Jenis Atribut (1)



- **Simple / Atomic Attribute:** adalah atribut yang tidak dapat dibagi-bagi lagi menjadi atribut yang lebih mendasar.
- **Composite Attribute:** atribut yang terdiri dari beberapa atribut yang lebih mendasar.

## Contoh:

- Atribut ALAMAT, terdiri atas atribut JALAN, KOTA, KODE\_POS.
- Atribut NAME, terdiri atas atribut FNAME, MNAME dan LNAME pada suatu entitas (EMPLOYEE).
- **Single-Valued Attribute:** atribut yang hanya memiliki satu harga/nilai.

## Contoh:

- Atribut UMUR pada entitas PEGAWAI
- Atribut LOCATIONS pada entitas DEPARTMENT

# Jenis-Jenis Atribut (2)



- **Multi-Valued Attribute:** adalah atribut yang memiliki isi lebih dari satu nilai.

## Contoh:

- Atribut PENDIDIKAN TINGGI pada entitas PEGAWAI, dapat berisi lebih dari satu nilai: SMP, SMU, Perguruan Tinggi (Sarjana), Doktor, dll.
  - Atribut HOBBY pada entitas MAHASISWA, dapat memiliki lebih dari satu nilai: sepak bola, menyanyi, menari, tennis, dsb.
  - Atribut PRASYARAT pada entitas MATA\_KULIAH, dapat memiliki lebih dari satu nilai: Konsep Pemrograman & Algoritma Struktur Data untuk prasyarat mata kuliah Pemrograman Lanjut.
- **Null Values Attribute:** adalah atribut dari entitas yang tidak memiliki nilai.

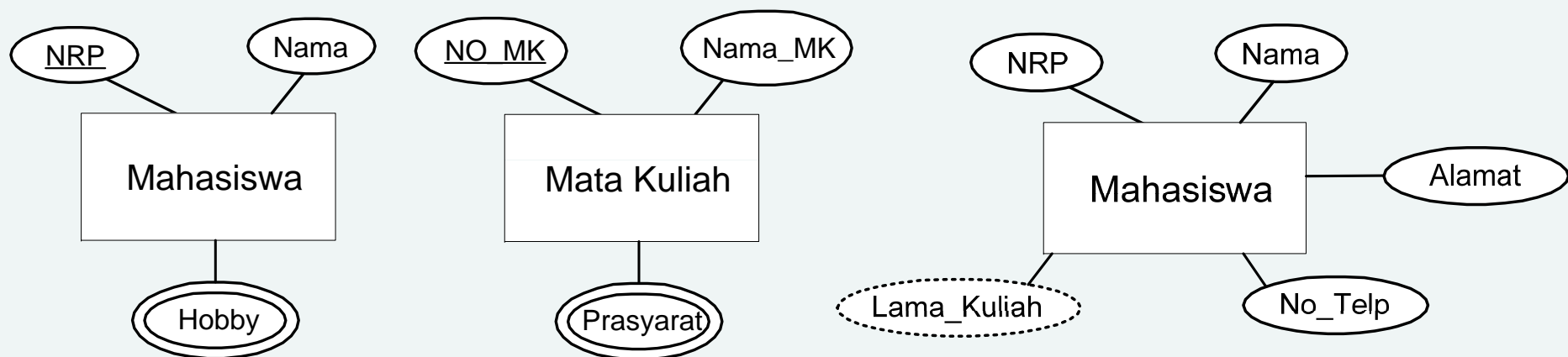
Contoh: Atribut PENDIDIKAN TINGGI untuk tamatan SMP.

# Jenis-Jenis Atribut (3)

- **Derived Attribute:** adalah atribut yang nilainya dapat diisi atau diturunkan dari perhitungan atau algoritma tertentu.

## Contoh:

- Atribut UMUR, dapat dihitung dari atribut TGL\_LAHIR
- Atribut LAMA\_KULIAH, dapat dihitung dari NIM yang merupakan kombinasi antara digit tahun dan digit yang lain (2696100...).
- Atribut INDEX\_PRESTASI, dapat dihitung dari NILAI yang diperoleh MAHASISWA.



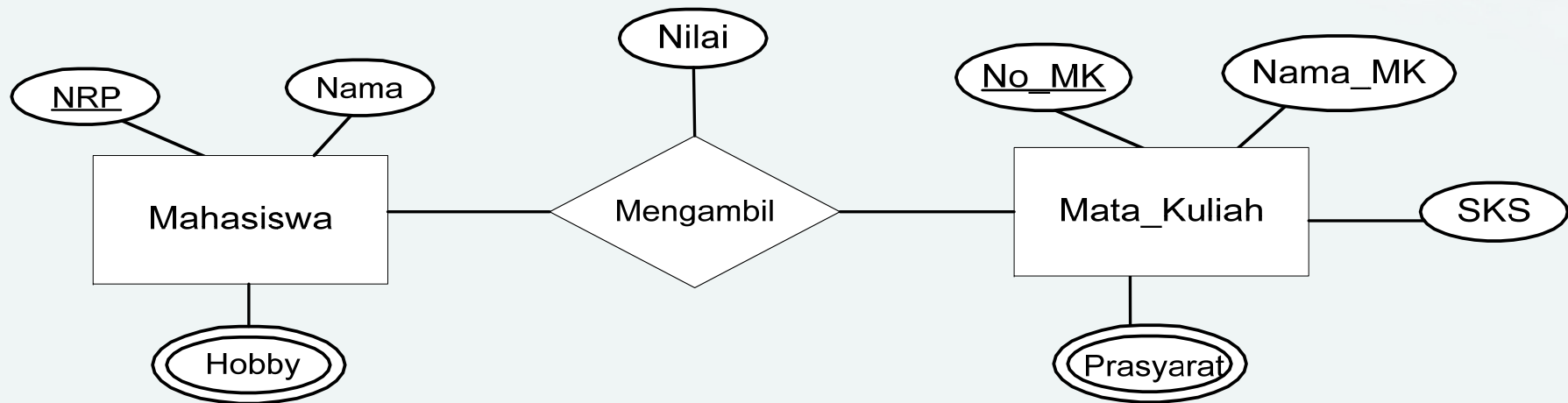
**Multi-Valued Attribute**

**Derived Attribute**



# Relasi dan Rasio Kardinalitas (1)

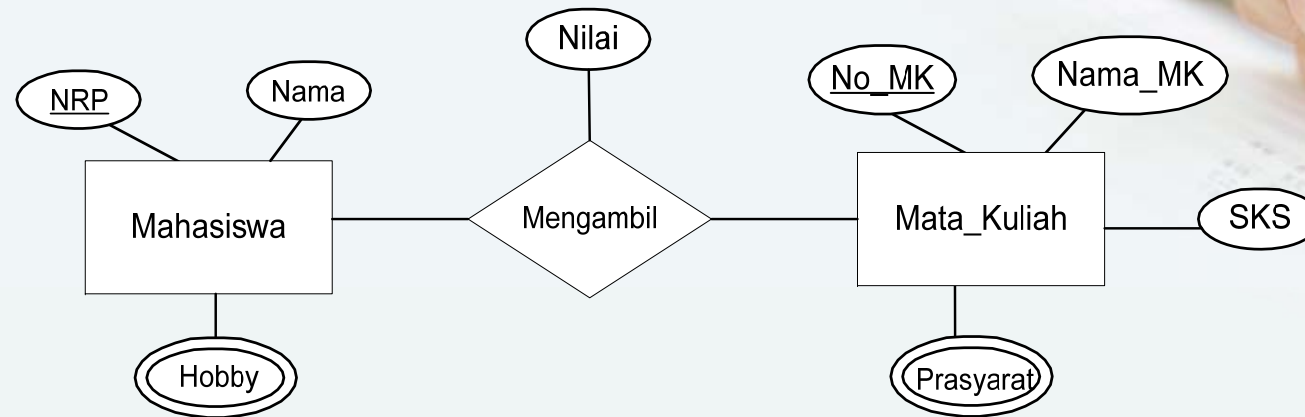
- **Relasi** adalah hubungan antar entitas.
- **Relasi** dapat memiliki **atribut**, dimana terjadi adanya transaksi yang menghasilkan suatu nilai tertentu.



## Penjelasan:

- Bentuk ER diatas antara Mahasiswa Mengambil Mata\_Kuliah, tentunya ada Nilai yang dihasilkan.
- Dimana atribut nilai ditempatkan?

# Relasi dan Rasio Kardinalitas (2)



## Penjelasan:

- Jika atribut **Nilai** ditempatkan pada entitas **Mahasiswa** (dimana **Nilai** merupakan salah satu atribut dari entitas **Mahasiswa**), maka semua mata kuliah yang diambil oleh seorang mahasiswa menghasilkan nilai yang sama (**tidak realistis**).
  - Jika atribut **Nilai** ditempatkan pada entitas **Mata\_Kuliah** (dimana **Nilai** merupakan salah satu atribut dari entitas **Mata\_Kuliah**), maka semua mahasiswa yang mengambil mata kuliah tertentu akan memiliki nilai yang sama (**tidak realistis**).
- Atribut **Nilai** harus ditempatkan pada relasi **Mengambil**, yang berarti seorang mahasiswa tertentu yang mengambil mata kuliah tertentu, akan mendapatkan nilai tertentu pula.

# Relasi dan Rasio Kardinalitas (3)

## Derajat Relasi

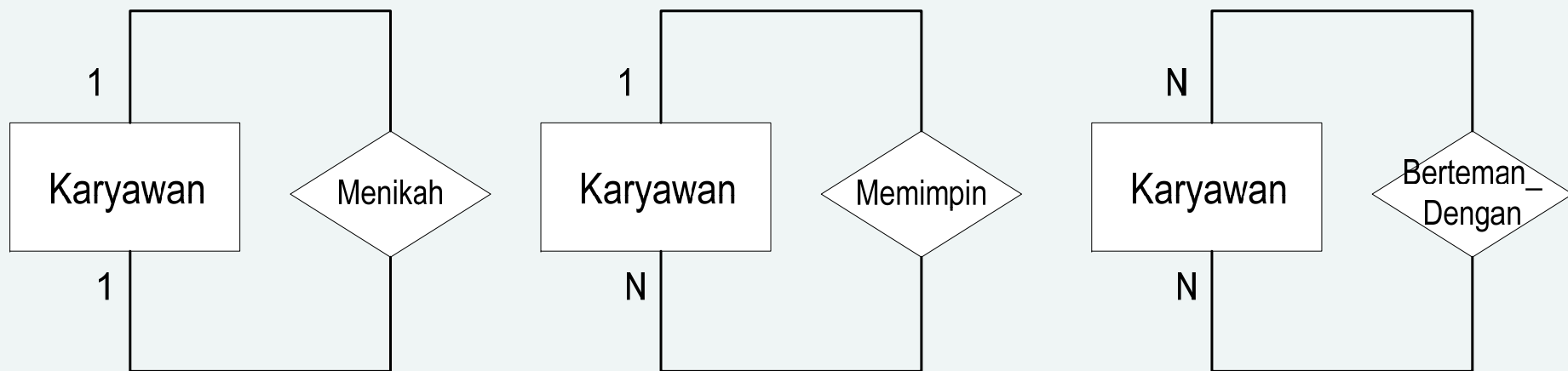
- **Derajat Relasi** adalah jumlah entitas yang berpartisipasi dalam suatu relasi.
- Derajat Relasi dapat berupa:
  - Unary Relationship (Relasi Berderajat 1)
  - Binary Relationship (Relasi Berderajat 2)
  - Ternary Relationship (Relasi Berderajat 3)

# Relasi dan Rasio Kardinalitas (4)

## Unary Relationship (Relasi Berderajat 1)

- adalah relasi dimana entitas yang terlibat hanya 1.
- Sering disebut relasi rekursif (recursive relationship).

Contoh:

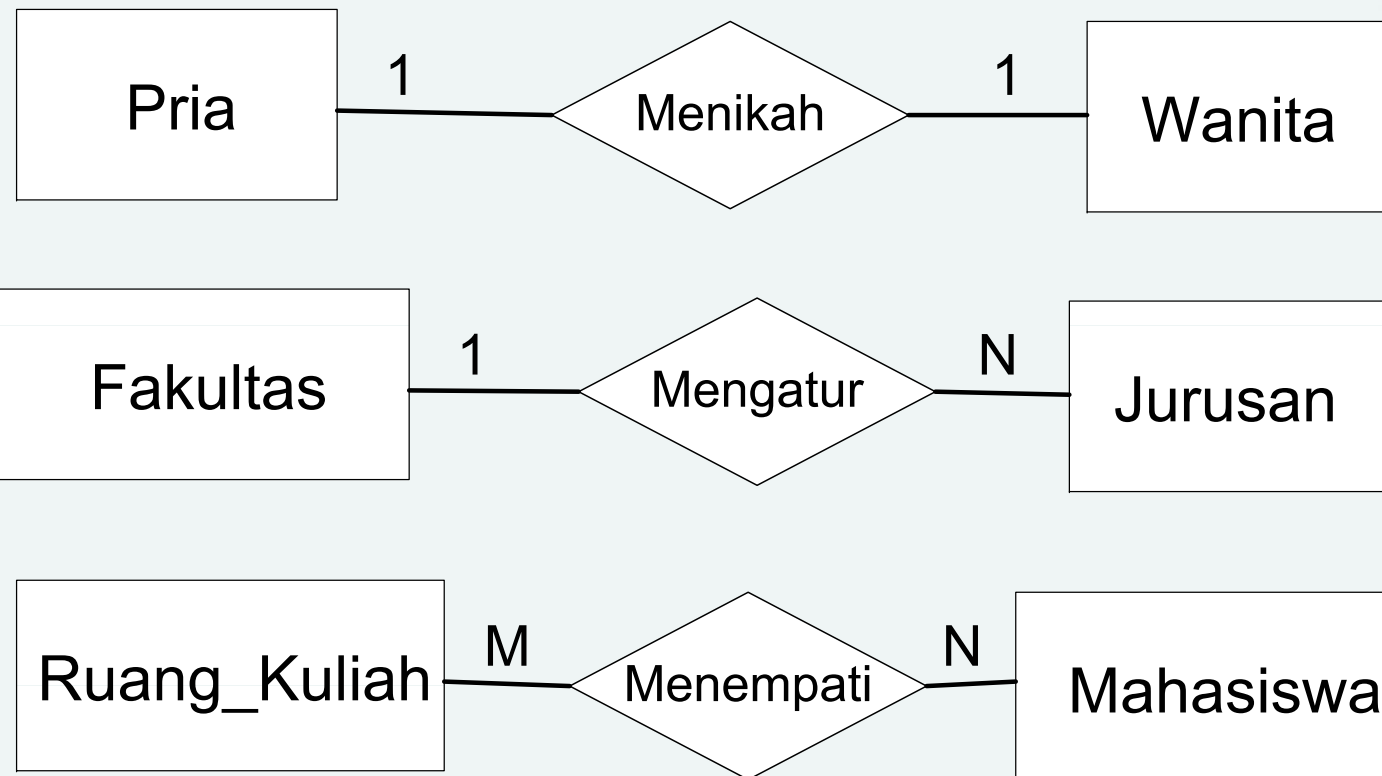


# Relasi dan Rasio Kardinalitas (5)

## Binary Relationship (Relasi Berderajat 2)

- Atau relasi **Biner** adalah relasi yang melibatkan 2 entitas.

Contoh:

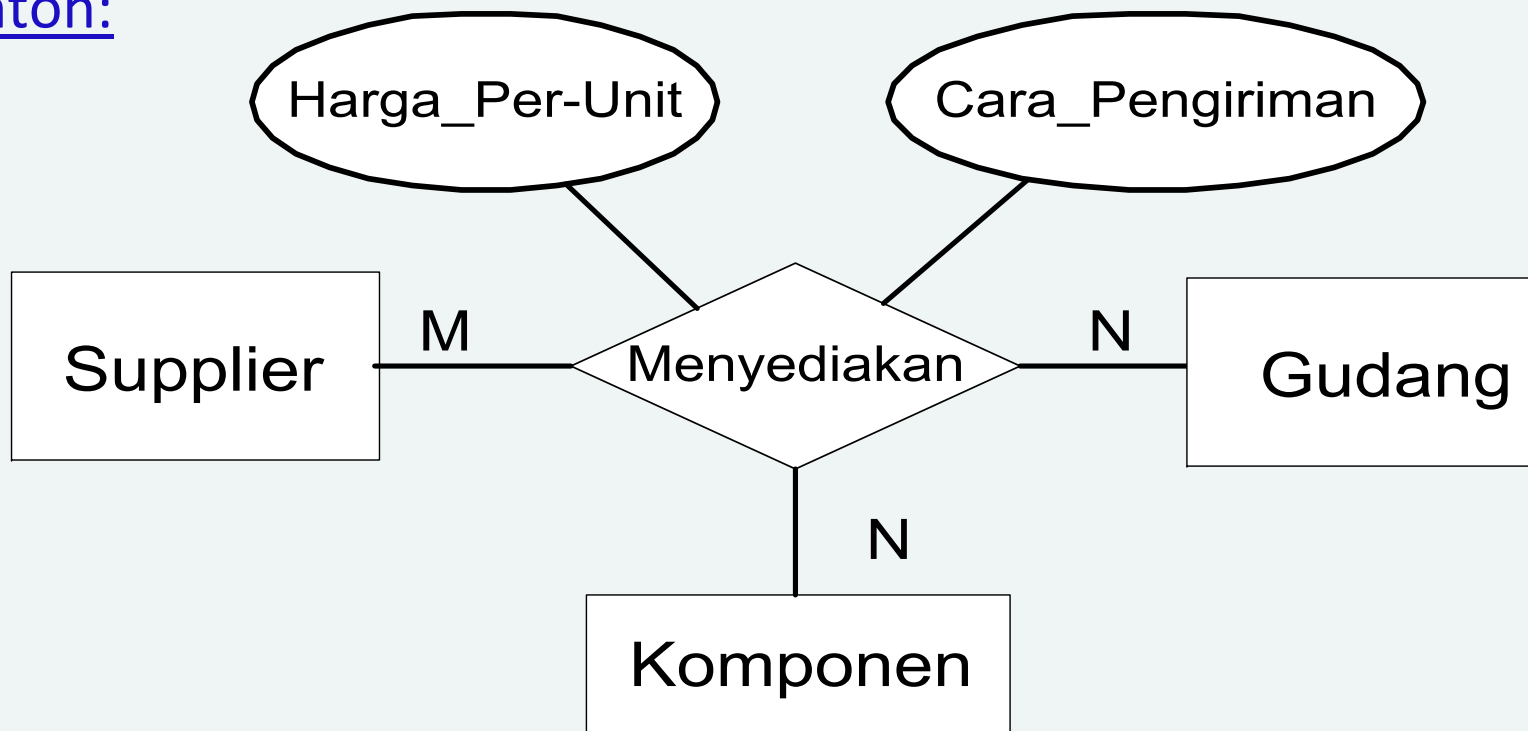


# Relasi dan Rasio Kardinalitas (6)

## Ternary Relationship (Relasi Berderajat 3)

- adalah relasi tunggal yang menghubungkan 3 entitas yang berbeda.

Contoh:

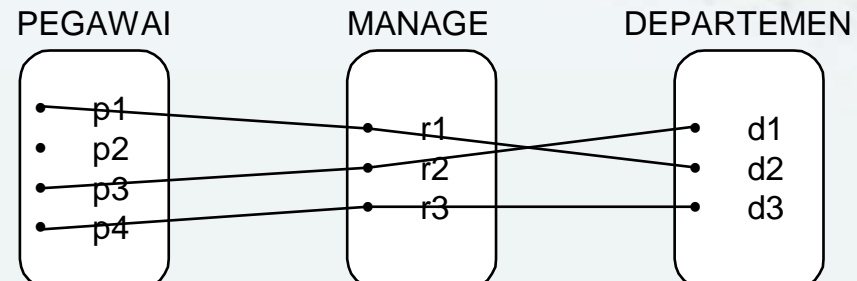


# Relasi dan Rasio Kardinalitas (7)

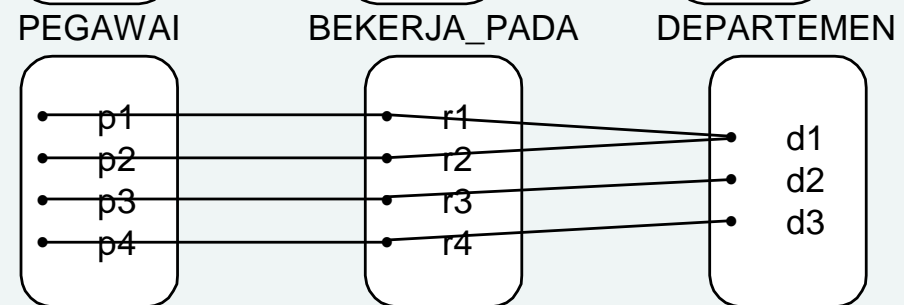
## Rasio Kardinalitas

- Dalam relasi binary antar 2 entitas, terdapat beberapa kemungkinan:

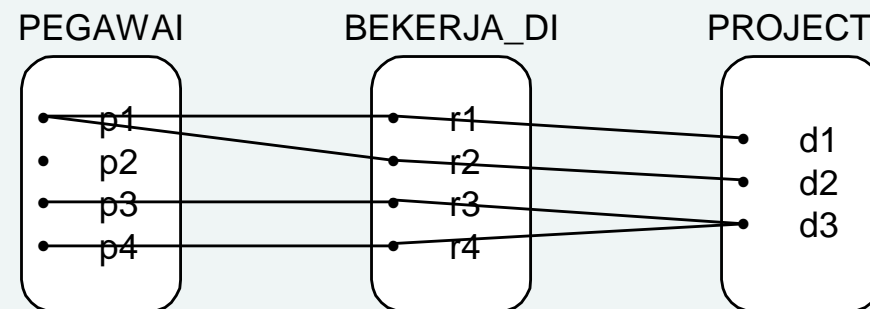
☑ **1 : 1 : One-to-One**



☑ **N : 1 : Many-to-One**



☑ **M : N : Many-to-Many**



# Participation Constraint Dependencies

- Menunjukkan apakah **keberadaan** suatu **entitas** bergantung **penuh / tidak** dengan entitas relasinya.
- Batasan (constraint) adalah jumlah **minimum** relasi dimana tiap entitas dapat ikut berpartisipasi.
- Ada **2 jenis Participation Constraint**:
  1. Partisipasi **Total ( === )**  
adalah bentuk partisipasi yang menunjukkan ketergantungan **penuh** suatu entitas (**semua dan harus**).
  2. Partisipasi **Parsial ( — )**  
adalah bentuk partisipasi yang menunjukkan ketergantungan **tidak penuh** suatu entitas (**beberapa, tidak harus semua**)

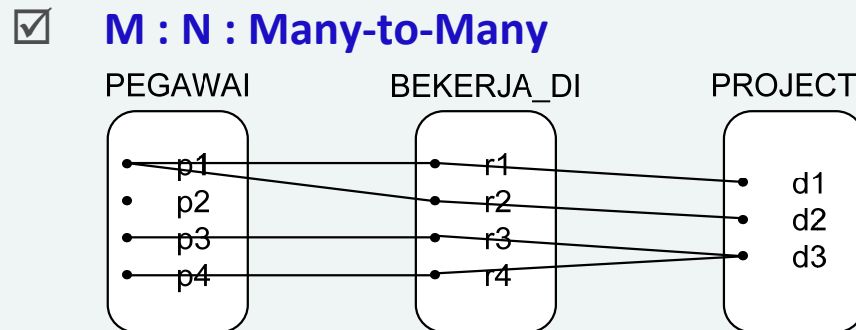
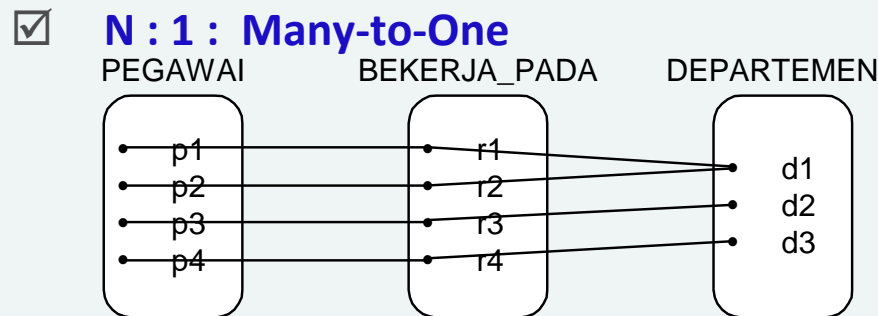
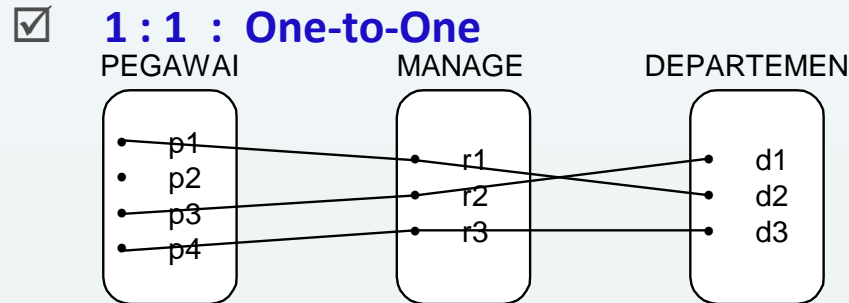




# Participation Constraint Dependencies



## Rasio Kardinalitas



## Participation Constraint



# Mapping ke Skema Relasi (1)

Untuk melakukan **mapping** (pemetaan) dari skema ER Diagram ke skema relasi terdapat langkah-langkah yang harus diperhatikan.

## Langkah-langkah mapping:

1. Untuk **setiap entitas** skema relasi **R** yang menyertakan seluruh **Simple Attribute** dan **Simple Attribute** dari **Composite Attribute** yang ada, pilih salah satu atribut kunci sebagai Primary Key.
2. Untuk setiap **Entitas Lemah**, buatlah skema relasi **R** dengan mengikutsertakan seluruh Simple Attribute. Tambahkan **Primary Key** dari **entitas kuatnya (Owner Entity type)** yang akan digunakan sebagai **Primary Key bersama-sama Partial Key** dari Entitas Lemah.

# Mapping ke Skema Relasi (2)



3. Untuk setiap relasi binary **1:1**, tambahkan **Primary Key** dari sisi yang lebih "**ringan**" ke sisi (entitas) yang lebih "**berat**". Suatu sisi dianggap lebih "**berat**" timbangannya apabila mempunyai **partisipasi total**. Tambahkan juga **Simple Attribute** yang terdapat pada relasi tersebut ke sisi yang lebih "**berat**".

Apabila kedua partisipasi adalah **sama total**, maka kedua entitas tersebut boleh **digabung** menjadi satu skema relasi.

4. Untuk setiap relasi binary **1:N** yang tidak melibatkan entitas lemah, tentukan mana sisi yang lebih "**berat**". Sisi dianggap lebih "**berat**" timbangannya adalah sisi-**N (Many)**. Tambahkan **Primary Key** dari sisi yang "**ringan**" ke skema relasi sisi yang lebih "**berat**". Tambahkan juga **seluruh simple attribute** yang terdapat pada relasi biner tersebut.

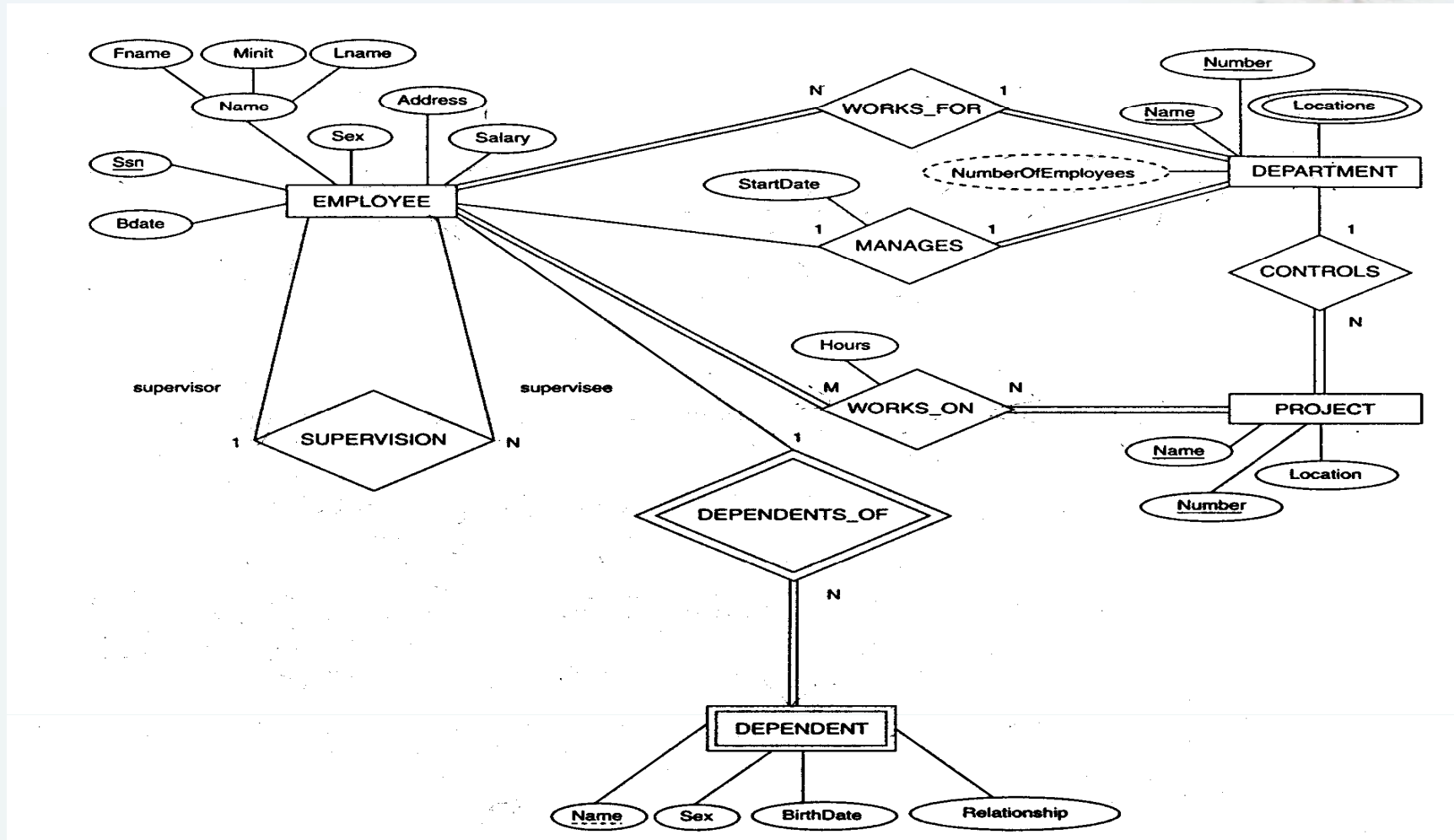
## Mapping ke Skema Relasi (3)



5. Untuk setiap relasi binary **M:N**, buatlah skema relasi baru **R** dengan atribut **seluruh simple attribute** yang terdapat pada relasi biner tersebut. Tambahkan **primary key** yang terdapat pada kedua sisi ke skema relasi R. Kedua **Foreign Key** yang didapat dari kedua sisi tersebut digabung menjadi satu membentuk **Primary Key** dari skema relasi R.
6. Untuk setiap **Multivalued Attribute**, buatlah skema relasi R yang menyertakan atribut dari multivalued tersebut. Tambahkan **Primary Key** dari relasi yang memiliki multivalued tersebut. Kedua atribut tersebut membentuk **Primary Key** dari skema relasi R.
7. Untuk setiap relasi n-ary dengan  $n > 2$ , buatlah skema relasi R yang menyertakan seluruh **Primary Key** dari entitas yang ikut serta. Sejumlah n **Foreign Key** tersebut akan membentuk **Primary Key** untuk skema relasi R. Tambahkan seluruh Simple Attribute yang terdapat pada relasi n-ary tersebut.

# Mapping ke Skema Relasi (4)

Diagram Skema Konseptual / ER Diagram untuk Database COMPANY



# Mapping ke Skema Relasi (5)

## EMPLOYEE

FNAME	MINIT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
-------	-------	-------	------------	-------	---------	-----	--------	----------	-----

## DEPARTMENT

DNAME	<u>DNUMBER</u>	MGRSSN	MGRSTARTDATE
-------	----------------	--------	--------------

## DEPT\_LOCATIONS

<u>DNUMBER</u>	<u>DLOCATION</u>
----------------	------------------

## PROJECT

PNAME	<u>PNUMBER</u>	LOCATION	DNUM
-------	----------------	----------	------

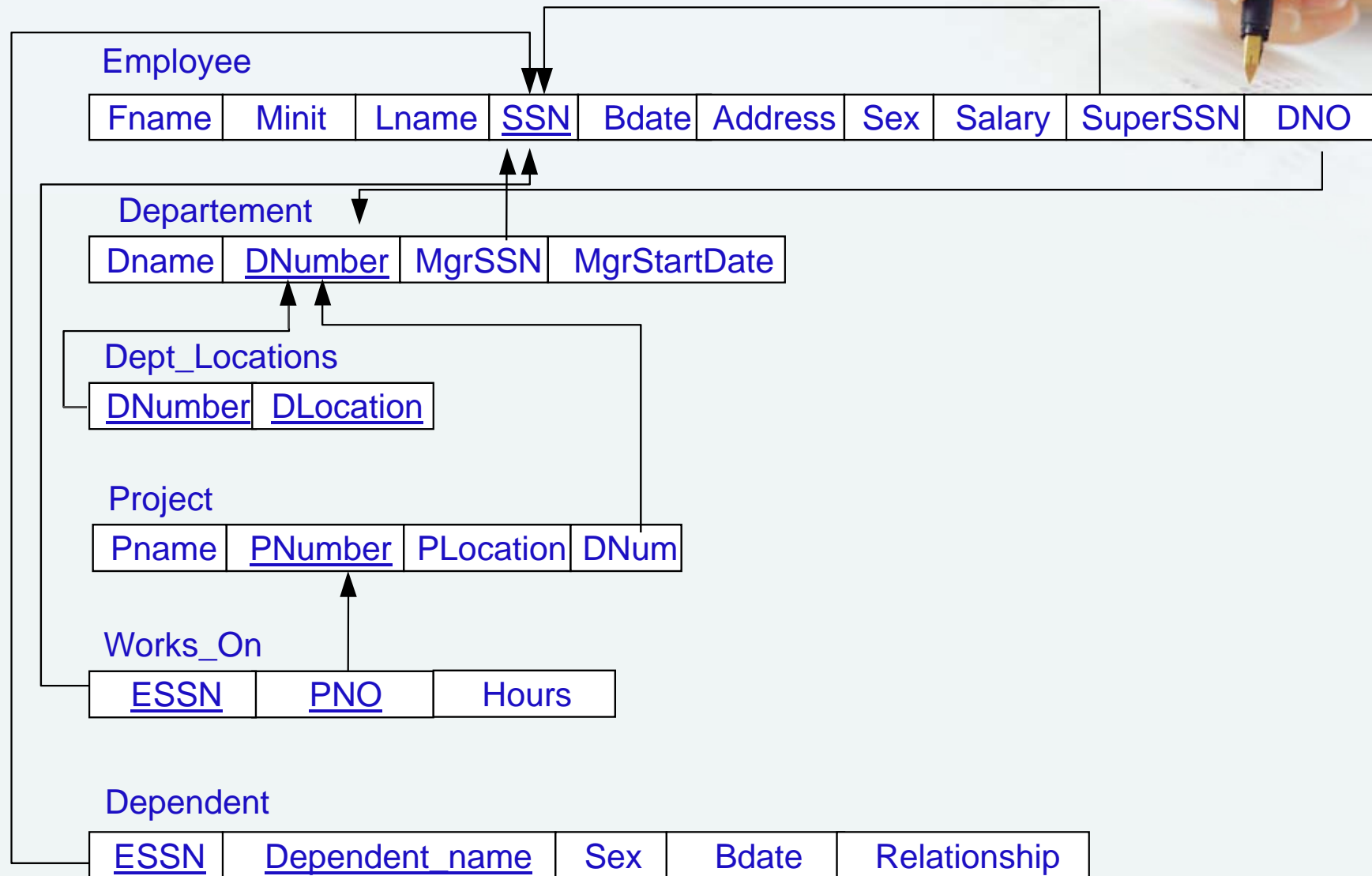
## WORKS ON

<u>ESSN</u>	<u>PNO</u>	HOURS
-------------	------------	-------

## DEPENDENT

<u>SSN</u>	<u>DEPENDENT_NAME</u>	SEX	BDATE	RELATIONSHIP
------------	-----------------------	-----	-------	--------------

# Mapping Skema ER Diagram dengan Referential Integrity Constraint



# Contoh Implementasi Data Table

## DEPARTMENT

dname	dnumber	mgrssn	mgrstartdate
Headquarters	1	333445555	22-May-78
administration	4	987654321	01-Jan-85
Research	5	888665555	19-Jun-71

## DEPARTMENT LOCATION

dnumber	dlocations
1	Houston
4	Stafford
5	Bellaire
5	Houston
5	Sugarland

## WORKS ON

essn	pno	hours
123456789	1	32.5
123456789	2	7.5
333445555	10	10
333445555	2	10
333445555	20	10
333445555	3	10
453453453	1	20
453453453	2	20
666884444	3	40
888665555	20	0
987654321	20	15
987654321	30	20
999887777	10	10
999887777	30	30



# Contoh Implementasi Data Table



## DEPENDENT

essn	dependent_name	sex	bdate	relationship
123456789	Alice	F	31-Dec-78	DAUGHTER
123456789	Elizabeth	F	05-May-57	SPOUSE
123456789	Michael	M	01-Jan-78	SON
333445555	Alice	F	05-Apr-76	DAUGHTER
333445555	Joy	F	03-May-48	SPOUSE
333445555	Theodore	M	25-Oct-73	SON
987654321	abner	M	29-Feb-32	SPOUSE

## EMPLOYEE

fname	minit	lname	ssn	bdate	address	sex	salary	superssn	dno
Ahmad	V	Jabbar	987987987	29-Mar-59	980 Dallas, Houston, TX	M	25000	987654321	4
Alicia	J	Zelaya	999887777	19-Jul-58	3321 Castle, Spring, TX	F	25000	987654321	4
Franklin	T	Wong	333445555	08-Dec-45	638 Voss, Houston, TX	M	40000	888665555	5
James	E	Borg	888665555	10-Nov-27	450 Stone, Houston, TX	M	55000		1
Jennifer	S	Wallace	987654321	20-Jun-31	291 Berry, Bellaire, TX	F	43000	888665555	4
John	B	Smith	123456789	01-Sep-55	731 Fondren, Houston, TX	M	30000	333445555	5
Joyce	A	English	453453453	31-Jul-62	5631 Rice, Houston, TX	F			
Ramesh	K	Narayan	666884444	15-Sep-52	975 Fire Oak, Humble, TX	F			

## PROJECT

pname	pnumber	plocation	dnum
ProductX	1	Bellaire	5
Computerization	10	Stafford	4
ProductY	2	Sugarland	5
Reorganization	20	Houston	1
ProductZ	3	Houston	5
Newbenefits	30	Stafford	4