

Database Analysis and Design

Lesson 7: Entity Relationship Model to Relational Model

Ms. SEAK LENG

Plan

- Map regular entities
- Map weak entities
- Map binary relationships
- Map associative entities
- Map unary relationships
- Map ternary relationships
- Map supertype/subtype relationships

Keywords

Keyword	Description
Primary Key	A Primary Key is a column (or a set of columns) in a table that uniquely identifies each record in that table. Each table can have only one primary key, and the values in the primary key must be unique and non-null for every row in the table..
Foreign Key	A Foreign Key is a column (or a set of columns) in one table that refers to the Primary Key of another table. It is used to establish and enforce a link between the data in the two tables.
Surrogate Key	An artificial, meaningless, primary key made up by the database designer under certain limited conditions.
Composite Key	A Composite Key (also known as a Concatenated Key) is a primary key that consists of two or more columns combined to uniquely identify each record in a table.
Partial Key	A key that identifies a record in combination with another key, typically in a weak entity relationship.

Introduction

- The logical design takes the high-level model from the conceptual design and translates it into a structure that can be implemented in a specific DBMS. It defines the structure of the data in terms of tables, columns, and relationships but still avoids consideration of how the data will be physically stored.
- The logical design typically uses the relational model (for RDBMS) where data is represented in tables. Validate the conceptual model using the technique of normalization, to remove redundancies.

Introduction

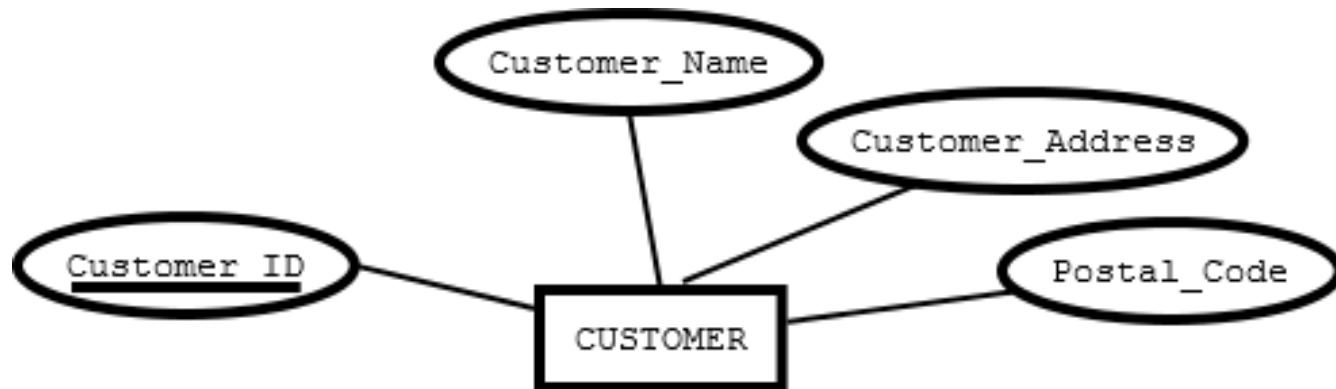
- Mapping an Entity-Relationship (ER) model to a Relational Model involves translating the conceptual schema represented in the ER diagram into a logical schema that can be implemented in a relational database.
- It involves systematically transforming the entities, attributes, and relationships in the ER diagram into relationship database structures such as tables, columns, and keys.

Mapping Regular Entities

- Mapping regular entities to tables
 - An entity set is mapped to a table in a straightforward way
- A). Simple attributes: E-R attributes are mapped directly into relation attributes.
 - each attribute of the entity set becomes an attribute/columns of the table.

Mapping Regular Entities

- Example: Mapping a regular entity with simple attributes
 - CUSTOMER entity type with simple attributes



- CUSTOMER relation



CUSTOMER			
<u>Customer_ID</u>	Customer_Name	Customer_Address	Postal_Code

CUSTOMER	
PK	<u>Customer_ID</u>
	Customer_Name
	Customer_Address
	Postal_Code

Mapping Regular Entities

- Example: Mapping a regular entity with simple attributes
 - CUSTOMER entity type with simple attributes
 - Example:

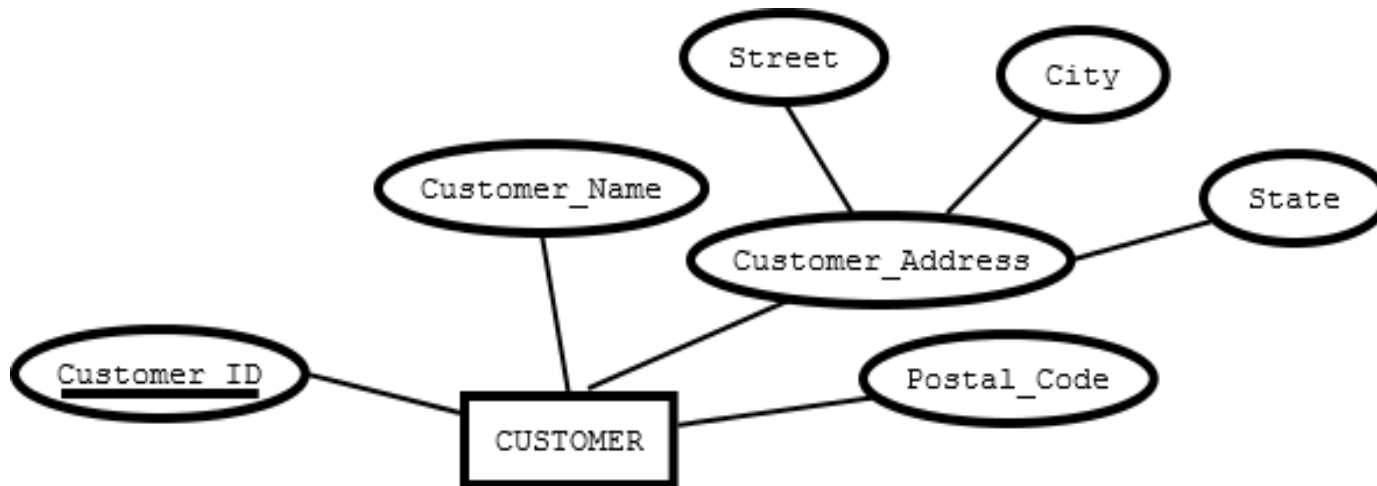
CUSTOMER	
PK	<u>Customer_ID</u>
	Customer_Name Customer_Address Postal_Code

CUSTOMER

<u>CustomerID</u>	CustomerName	Customer_Address	Postal_Code
1	John Doe	123 Maple Street, New York, NY, USA	10001
2	Jane Smith	456 Oak Avenue, Los Angeles, CA, USA	90001
3	Michael Brown	789 Pine Road, Chicago, IL, USA	60601
4	Emily Johnson	321 Cedar Lane, Houston, TX, USA	77001

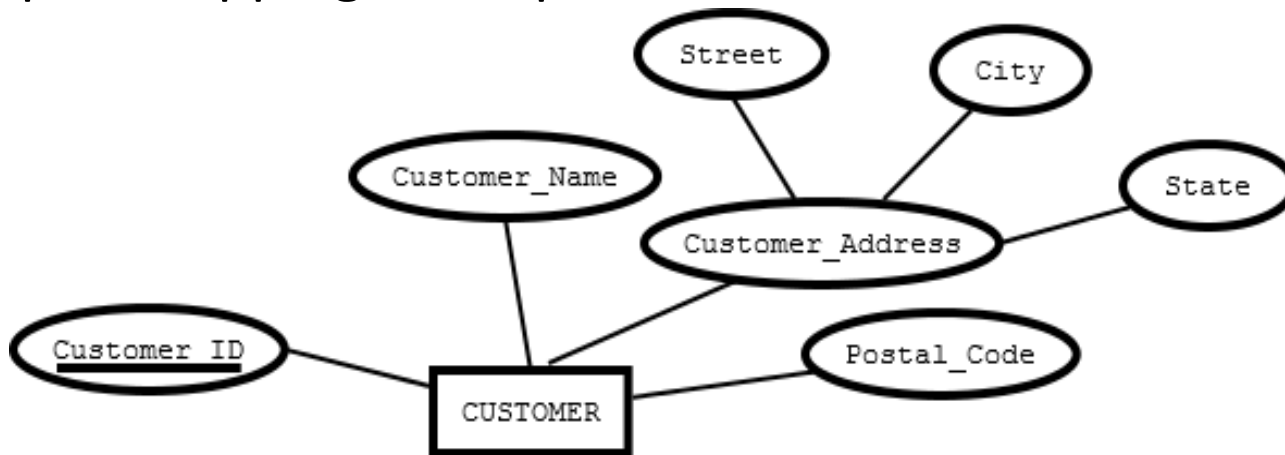
Mapping Regular Entities

- B). Composite attributes:
 - Decompose the composite attribute into individual attributes and possibly create a separate table for the entity if it is used in multiple places (e.g. user's billing and shipping addresses).
- Example: Mapping a composite attribute
 - CUSTOMER entity type with composite attribute



Mapping Regular Entities

- Example: Mapping a composite attribute



CUSTOMER relation with address detail

CUSTOMER

<u>Customer_ID</u>	Customer_Name	Street	City	State	Postal_Code
--------------------	---------------	--------	------	-------	-------------

CUSTOMER

PK	<u>Customer_ID</u>
	Customer_Name
	Street
	City
	State
	Postal_Code

Mapping Regular Entities

- Example: Mapping a composite attribute

– Example:

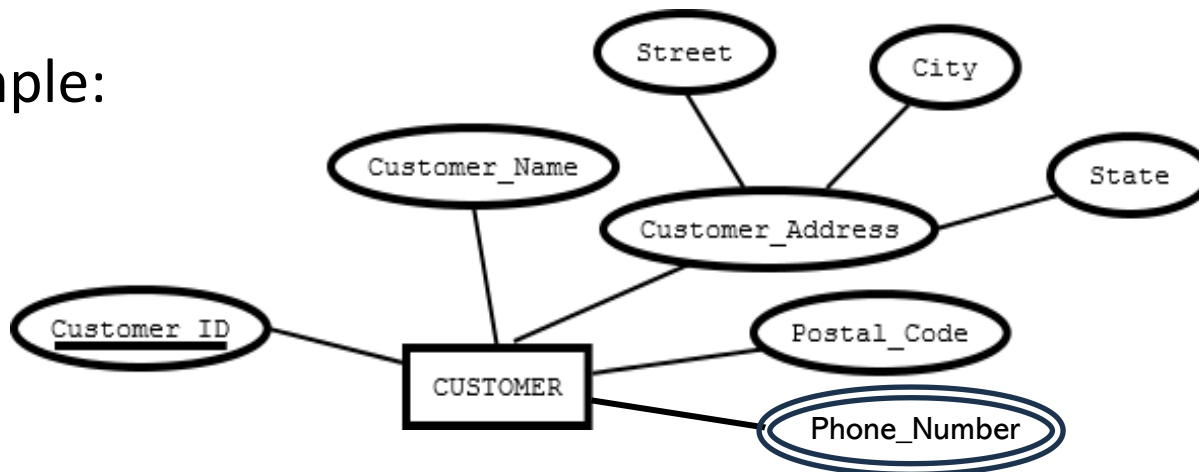
CUSTOMER	
PK	<u>Customer_ID</u>
	Customer_Name
	Street
	City
	State
	Postal_Code

<u>Customer_ID</u>	Customer_Name	Street	City	State	Postal_Code
1	John Doe	123 Maple Street	New York	NY	10001
2	Jane Smith	456 Oak Avenue	Los Angeles	CA	90001
3	Michael Brown	789 Pine Road	Chicago	IL	60601
4	Emily Johnson	321 Cedar Lane	Houston	TX	77001

Mapping Regular Entities

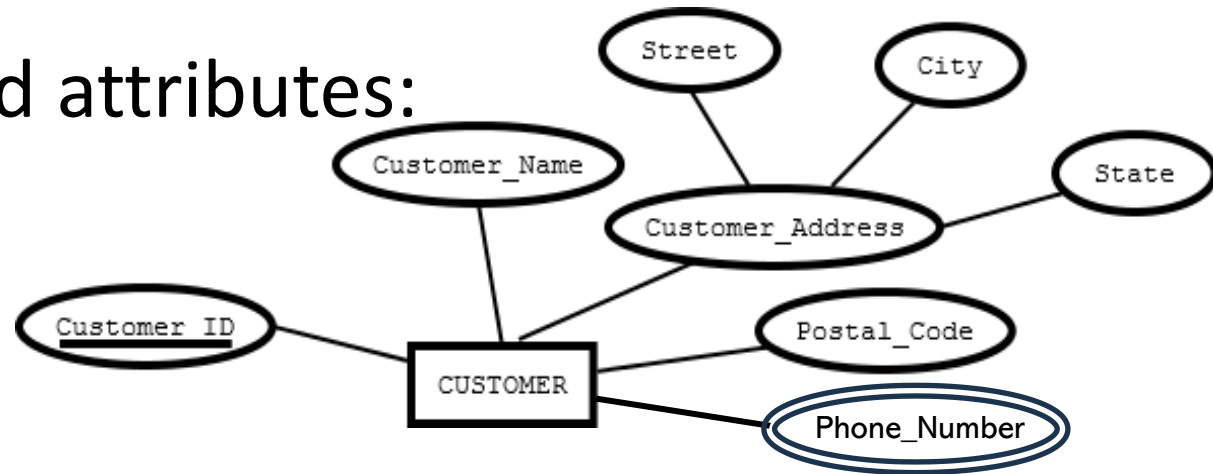
- C). Multivalued attributes:
 - Create a new table for the multivalued attribute.
 - The table will include: A primary key (typically a composite key) consisting of the primary key of the original entity and the multivalued attribute.
 - Remove the multivalued attribute from the original entity

- Example:



Mapping Regular Entities

- C). Multivalued attributes:
- Example:



CUSTOMER					
<u>Customer_ID</u>	Customer_Name	Street	City	State	Postal_Code

CustomerPhoneNumber	
<u>Customer_ID</u>	<u>Phone_Number</u>

Composite primary key

CUSTOMER	
PK	<u>Customer_ID</u>
	Customer_Name
	Street
	City
	State
	Postal_Code

CustomerPhoneNumber	
PK, FK	<u>Customer_ID</u>
PK	<u>Phone_Number</u>
	13

Mapping Regular Entities

- C). Multivalued attributes:
- Example:

CUSTOMER

<u>Customer_ID</u>	Customer_Name	Street	City	State	Postal_Code
1	John Doe	123 Maple Street	New York	NY	10001
2	Jane Smith	456 Oak Avenue	Los Angeles	CA	90001
3	Michael Brown	789 Pine Road	Chicago	IL	60601
4	Emily Johnson	321 Cedar Lane	Houston	TX	77001

CustomerPhoneNumber

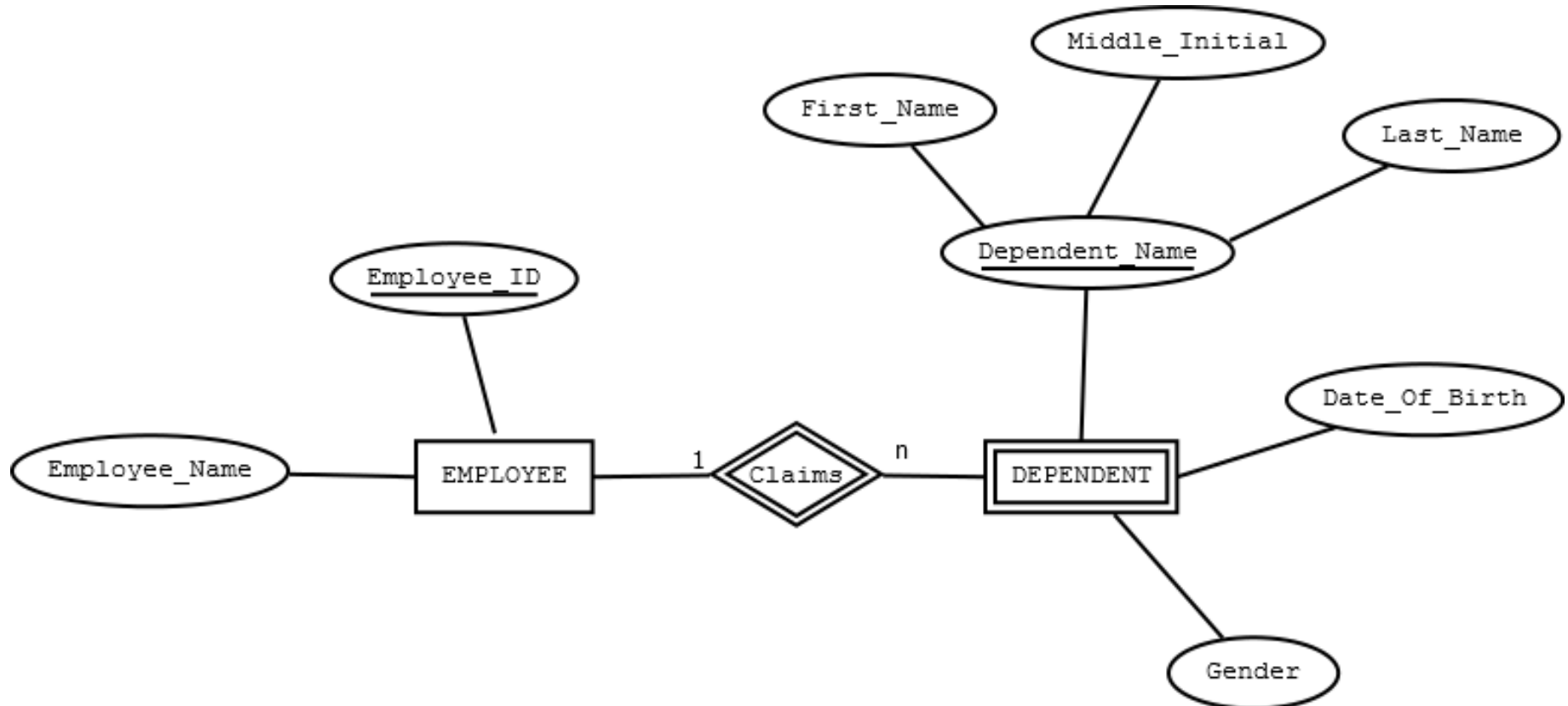
<u>Customer_ID</u>	<u>Customer_Name</u>
1	011-555-1234
1	012-555-5678
2	011-555-9876
2	010-555-2345
2	016-555-8765
3	099-555-4321

Mapping Weak Entities

- Mapping Weak Entities
 - It becomes a separate relation with a foreign key taken from the owner entity.
 - A weak entity needs a composite key including a foreign key from its parent. Its primary key is composed of:
 - Partial identifier of weak entity.
 - Primary key of identifying relation (strong entity).
 - Rename attributes, if necessary to avoid name conflicts.

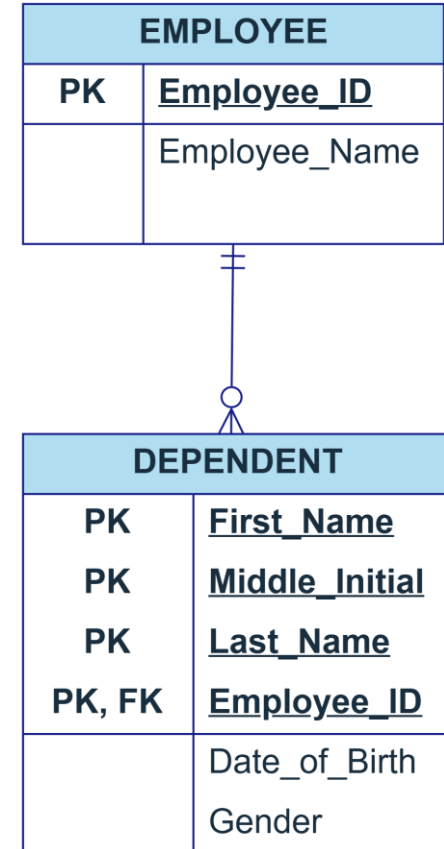
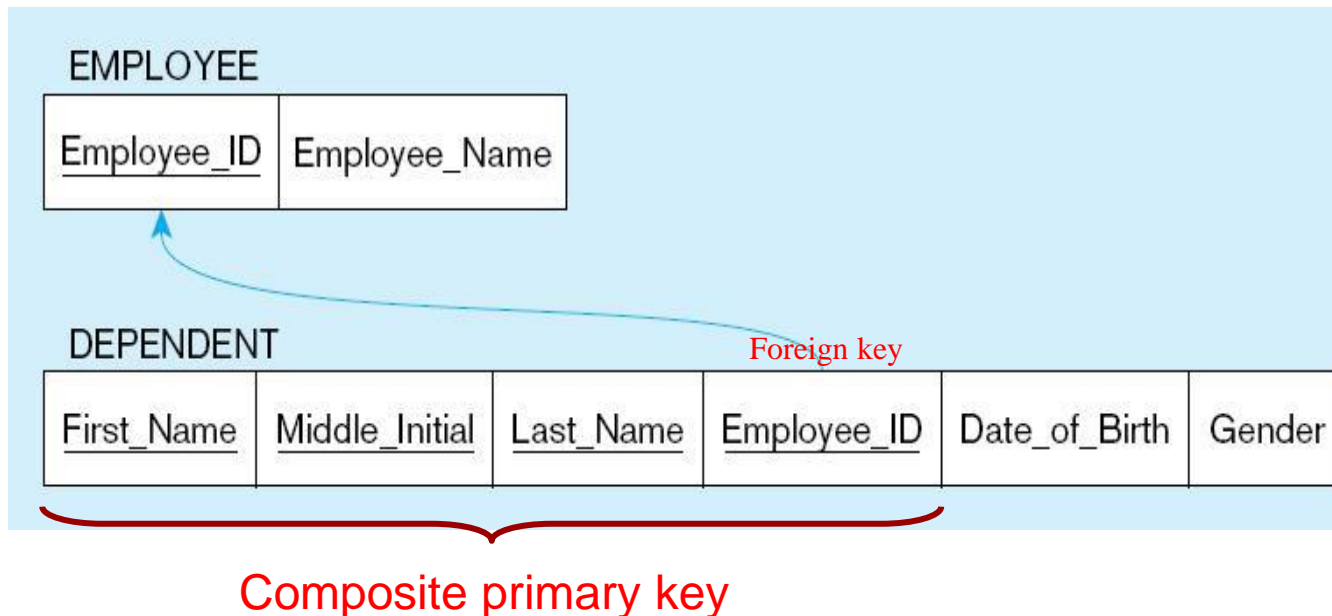
Mapping Weak Entities

- Example:
 - Weak entity DEPENDENT



Mapping Weak Entities

- Example:
 - Relations resulting from weak entity



NOTE: the domain constraint for the foreign key **Employee_ID** should NOT allow *null* value if **DEPENDENT** is a weak entity.

Mapping Weak Entities

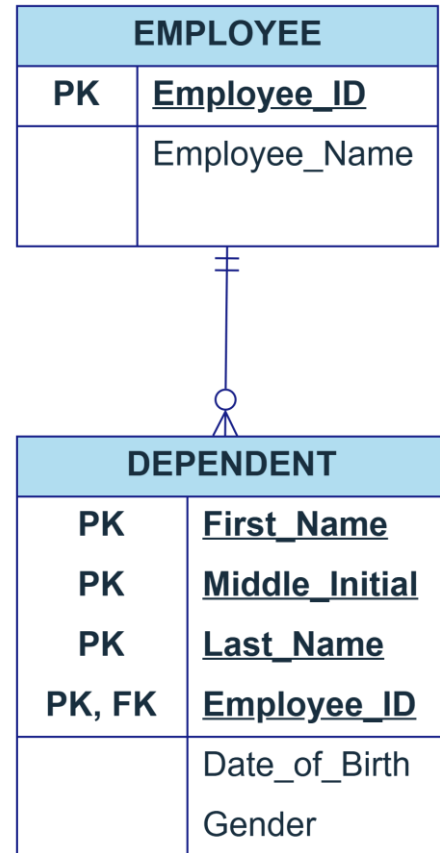
- Example:
 - Relations resulting from weak entity

EMPLOYEE

<u>Employee_ID</u>	Employee_Name
1	John Smith
2	Jane Doe
3	Michael Brown

DEPENDENT

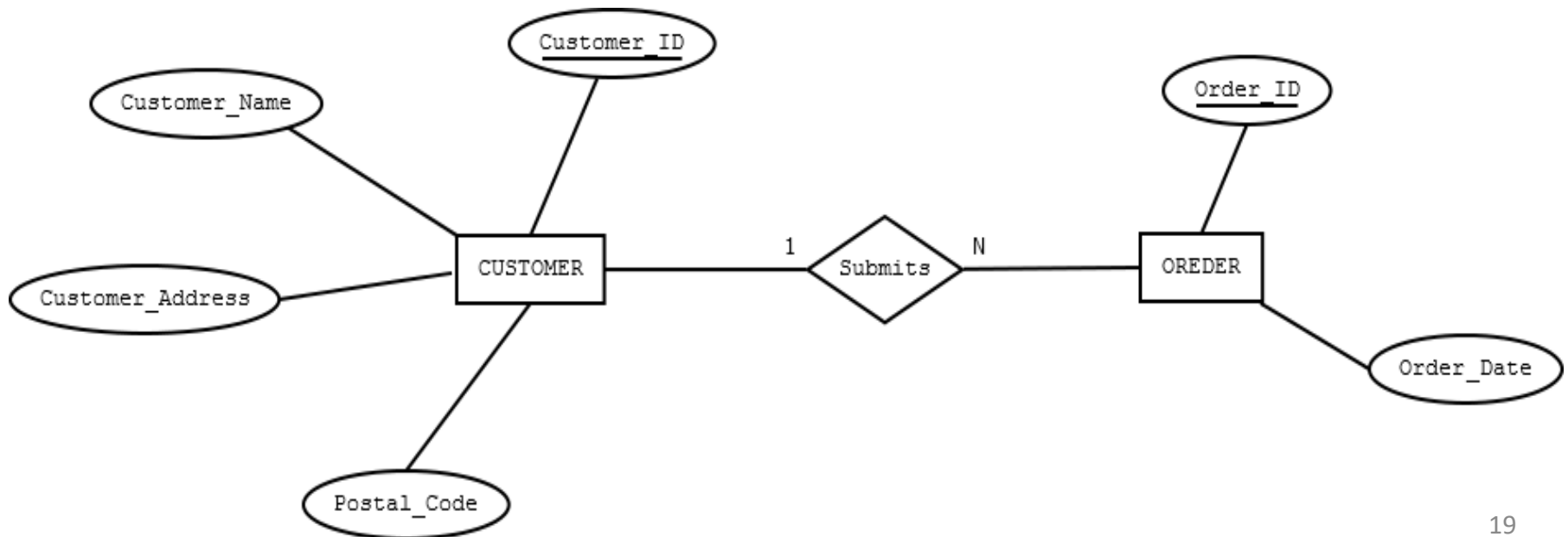
<u>First_Name</u>	<u>Middle_Initial</u>	<u>Last_Name</u>	<u>Employee_ID</u>	Date_of_Birth	Gender
Emily	A	Smith	1	5/10/2012	Female
Jack	B	Smith	1	8/23/2015	Male
Sarah	C	Doe	2	11/30/2010	Female



NOTE: the domain constraint for the foreign key Employee_ID should NOT allow *null* value if DEPENDENT is a weak entity.

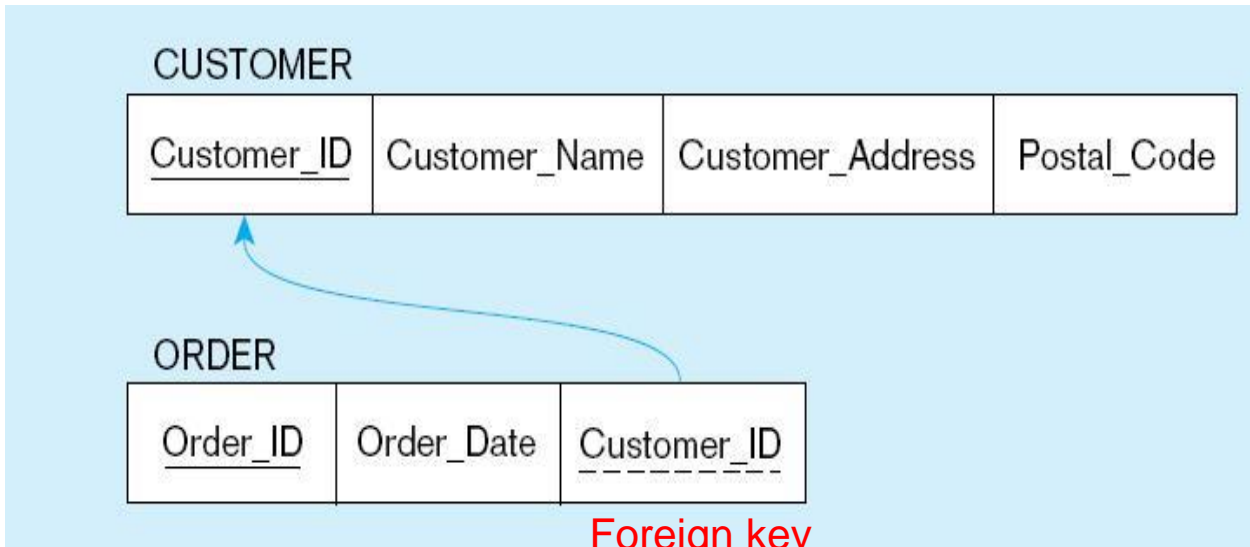
Mapping Binary Relationships

- One-to-Many: Primary key on the one side becomes a foreign key on the many side.
 - Example : Mapping a 1:N relationship
Relationship between customers and orders

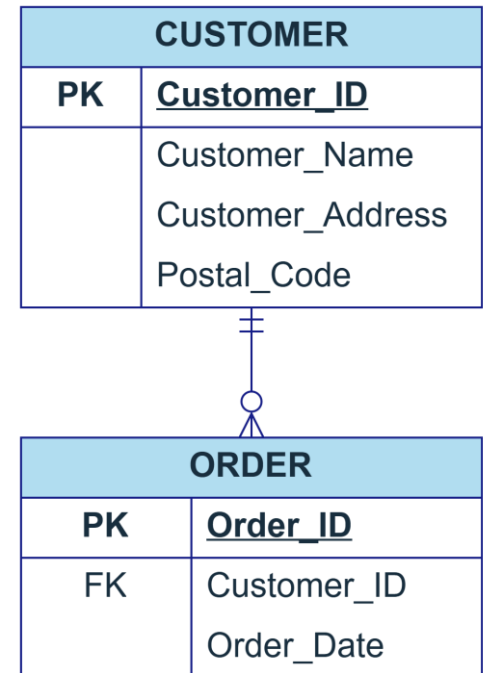


Mapping Binary Relationships

- Example : Mapping a 1:N relationship
 - Mapping the 1:N relationship



Again, no null value in the foreign key...this is because of the mandatory minimum cardinality



Mapping Binary Relationships

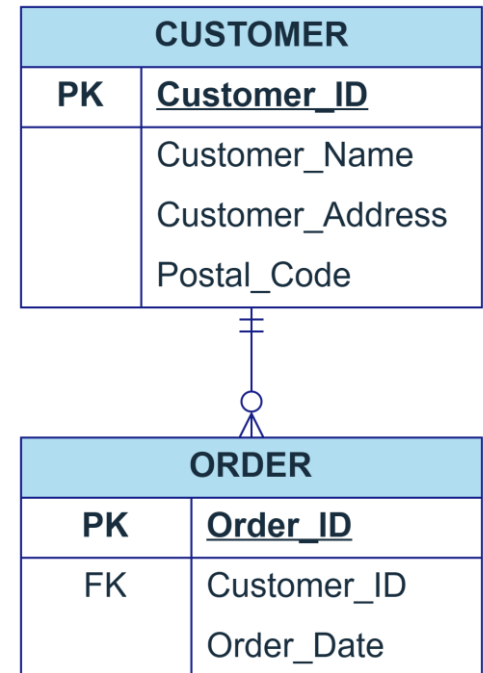
- Example : Mapping a 1:N relationship
 - Mapping the 1:N relationship

CUSTOMER

<u>CustomerID</u>	CustomerName	Customer_Address	Postal_Code
1	John Doe	123 Maple Street, New York, NY, USA	10001
2	Jane Smith	456 Oak Avenue, Los Angeles, CA, USA	90001
3	Michael Brown	789 Pine Road, Chicago, IL, USA	60601
4	Emily Johnson	321 Cedar Lane, Houston, TX, USA	77001

ORDER

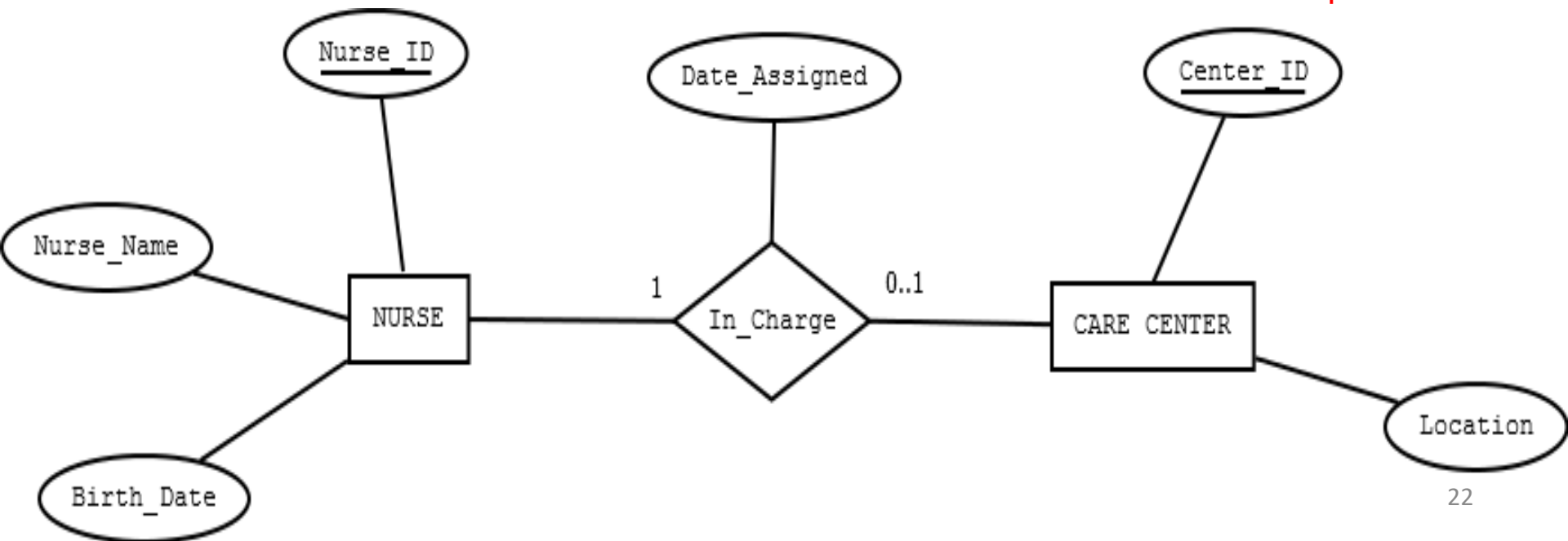
<u>Order_ID</u>	<u>Customer_ID</u>	Order_Date
101	1	11/1/2024
102	2	11/3/2024
103	3	11/5/2024
104	1	11/7/2024



Mapping Binary Relationships

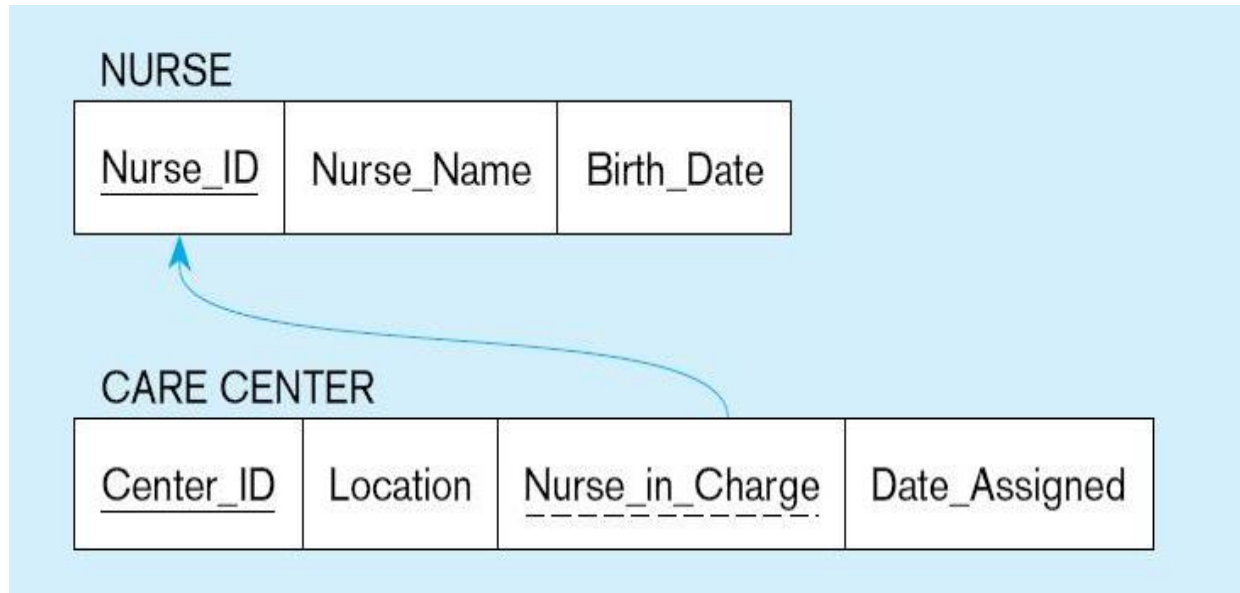
- One-to-One: Primary key on the mandatory side becomes a foreign key on the optional side.
 - Example : Mapping a 1:1 relationship
 - In_charge relationship (1:1)

Often in 1:1 relationships,
one direction is optional

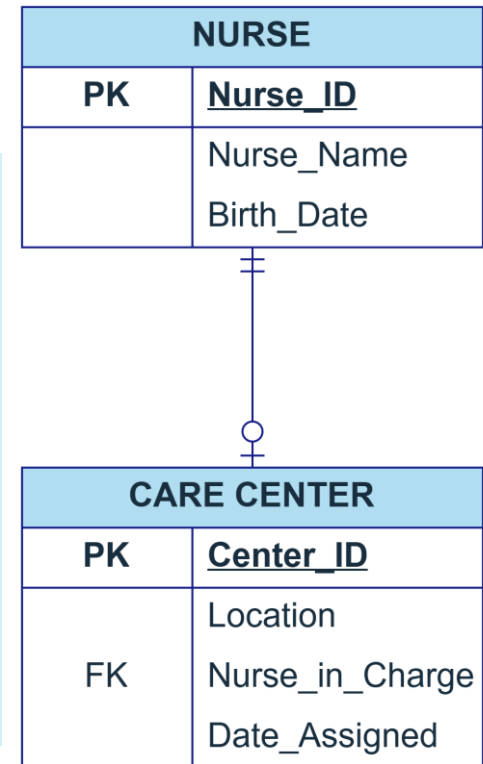


Mapping Binary Relationships

- Example : Mapping a 1:1 relationship
 - Resulting relations



Foreign key goes in the relation on the optional side, matching the primary key on the mandatory side



Mapping Binary Relationships

- Example : Mapping a 1:1 relationship
 - Resulting relations

NURSE

<u>Nurse_ID</u>	Nurse_Name	Birth_Date
1	Alice Johnson	3/12/1985
2	Bob Williams	7/21/1990
3	Clara Evans	11/5/1978
4	David Miller	2/19/1982

CARE CENTER

<u>Center_ID</u>	Location	Nurse_in_Charge	Date_Assigned
101	Central Health Center	1	6/15/2024
102	Eastside Care Facility	2	12/1/2023
103	Westview Rehabilitation	3	1/10/2024

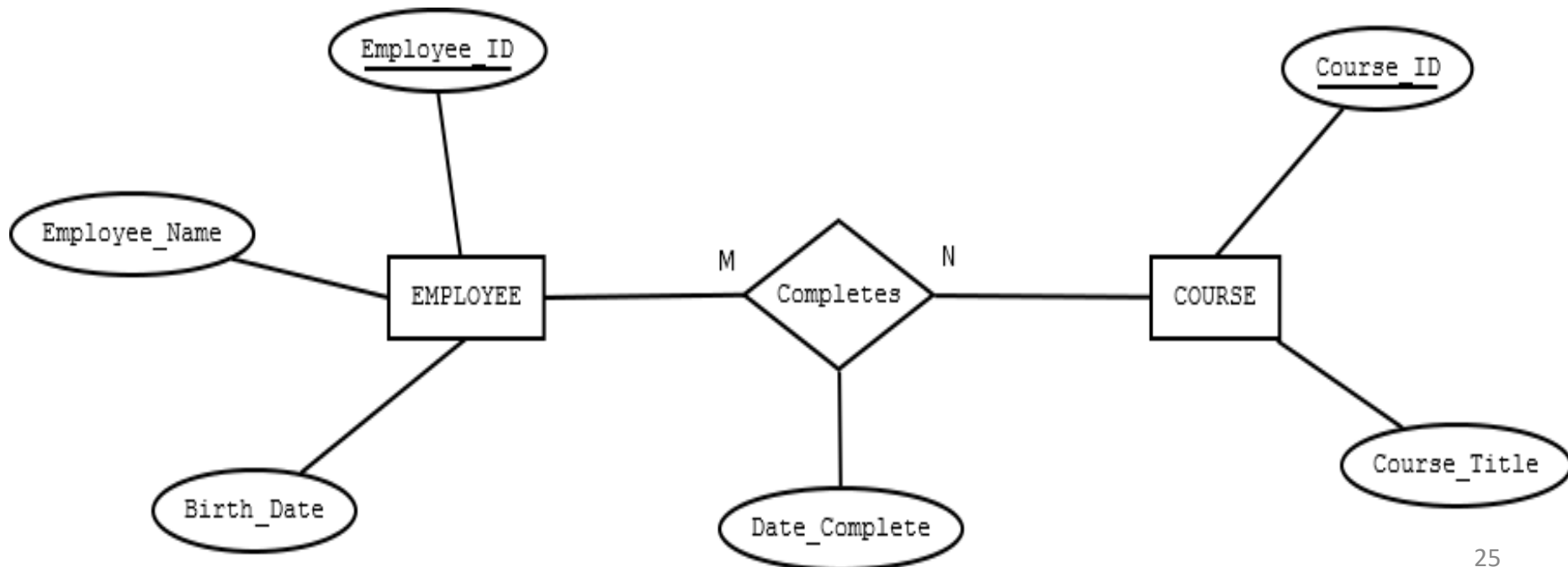
NURSE	
PK	<u>Nurse_ID</u>
	Nurse_Name
	Birth_Date



CARE CENTER	
PK	<u>Center_ID</u>
	Location
FK	Nurse_in_Charge
	Date_Assigned

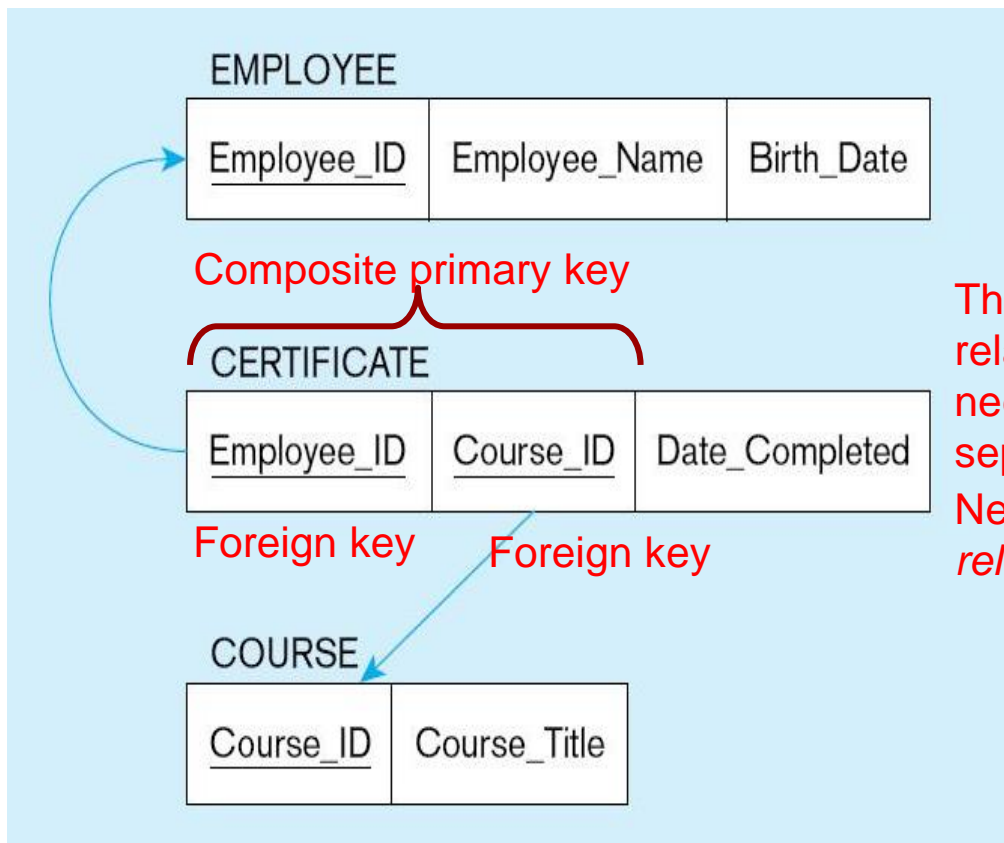
Mapping Binary Relationships

- Many-to-Many: Add a new relation with the primary keys of the two entities as its primary key. The new relation is called associative table.
 - Example 2: Mapping a M:N relationship

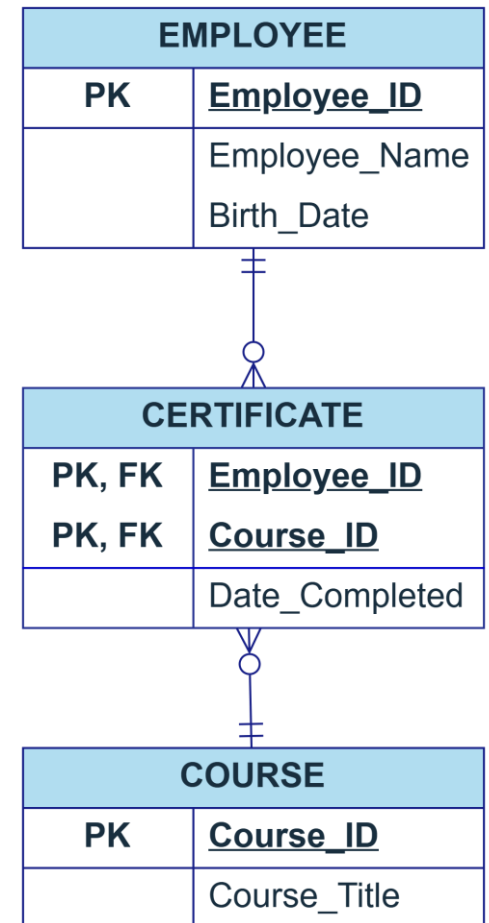


Mapping Binary Relationships

- Example 2: Mapping a M:N relationship
 - Three resulting relations



The Completes relationship will need to become a separate relation. New intersection relation



Mapping Binary Relationships

- Example 2: Mapping a M:N relationship
 - Three resulting relations

EMPLOYEE

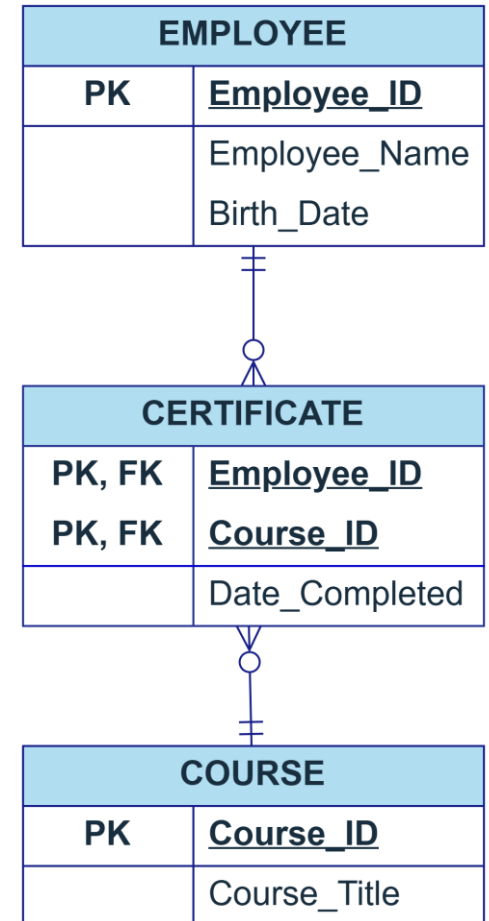
<u>Employee_ID</u>	Employee_Name	Birth_Date
1	John Smith	6/25/1980
2	Sarah Johnson	3/14/1990
3	Michael Brown	11/9/1985
4	Emma Williams	7/21/1992

CERTIFICATE

<u>Employee_ID</u>	<u>Course_ID</u>	Date_Completed
1	101	5/10/2023
1	102	2/20/2024
2	102	1/18/2024
3	103	12/1/2023

COURSE

<u>Course_ID</u>	Course_Title
101	Advanced SQL
102	Data Science Foundations
103	Cloud Computing Basics
104	Leadership and Management

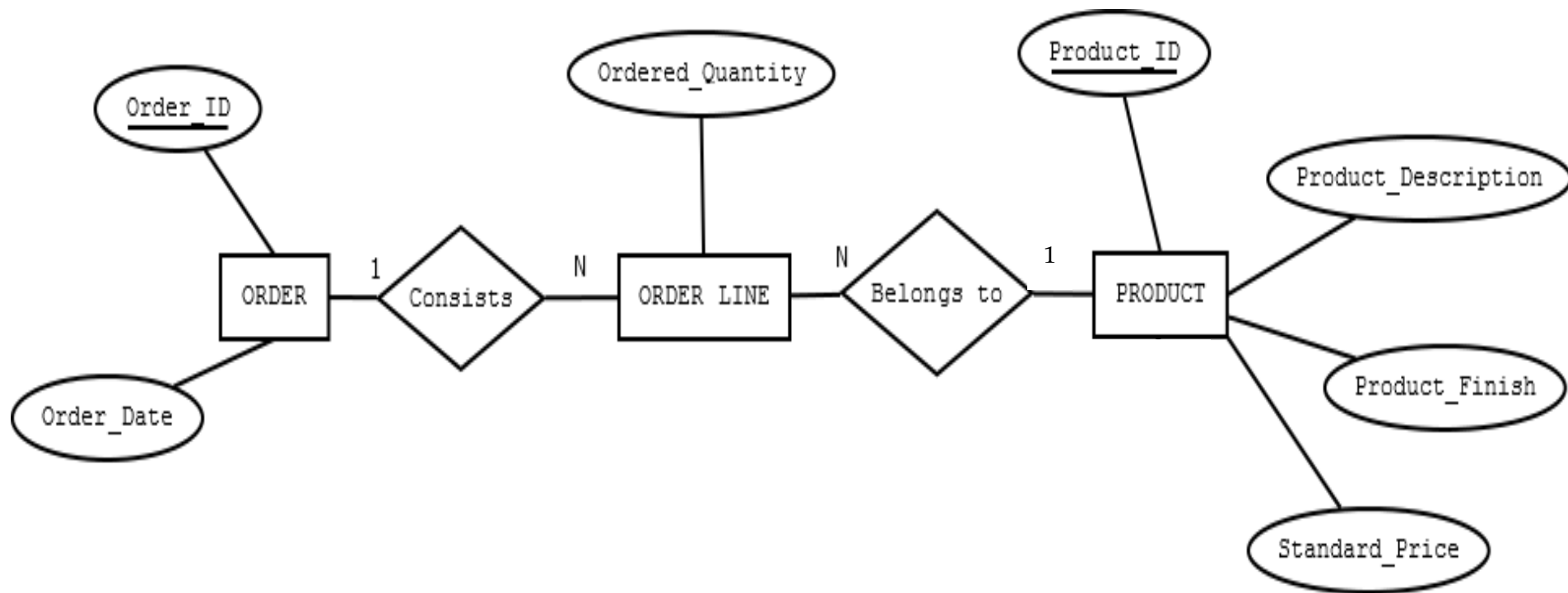


Mapping Associative Entities

- Mapping Associative Entities
 - a. Identifier is not assigned
 - Default primary key for the association relation is composed of the primary keys of the two entities (as in M:N relationship).
 - Example 1: Mapping an associative entity, no identifier assigned

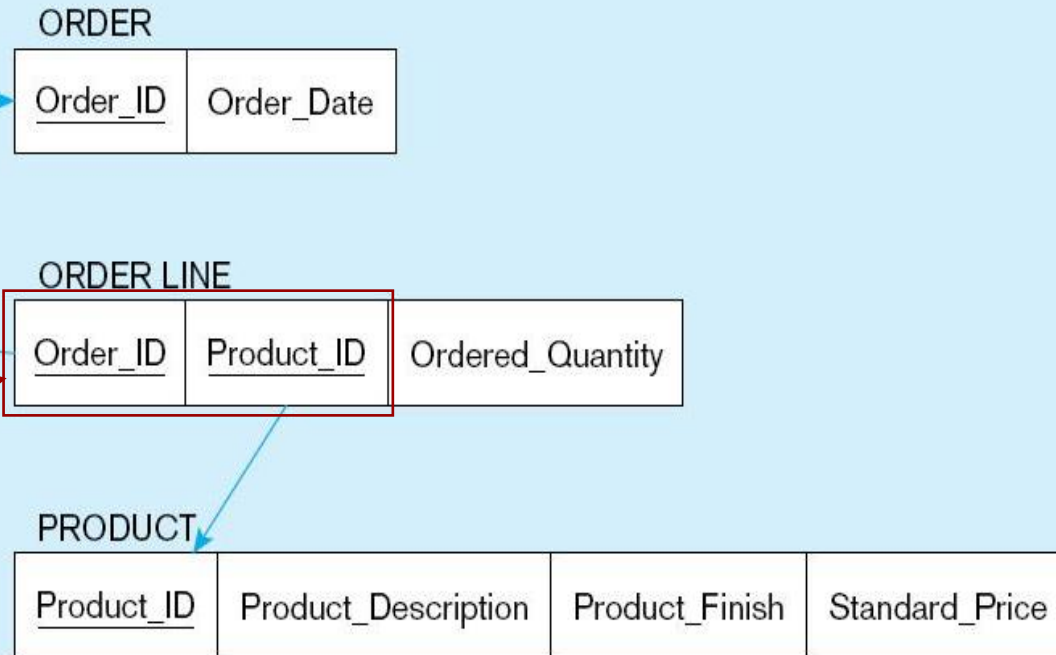
Mapping Associative Entities

- Example 1: Mapping an associative entity, no identifier assigned

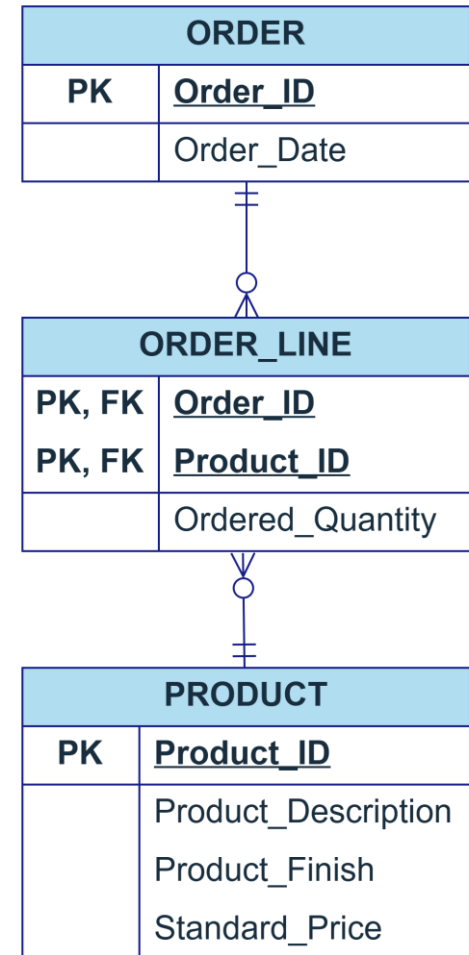


Mapping Associative Entities

- Example 1: Mapping an associative entity, no identifier assigned
 - Three resulting relations



Composite primary key formed from the two foreign keys



Mapping Associative Entities

- Example 1: Mapping an associative entity, no identifier assigned
 - Three resulting relations

ORDER

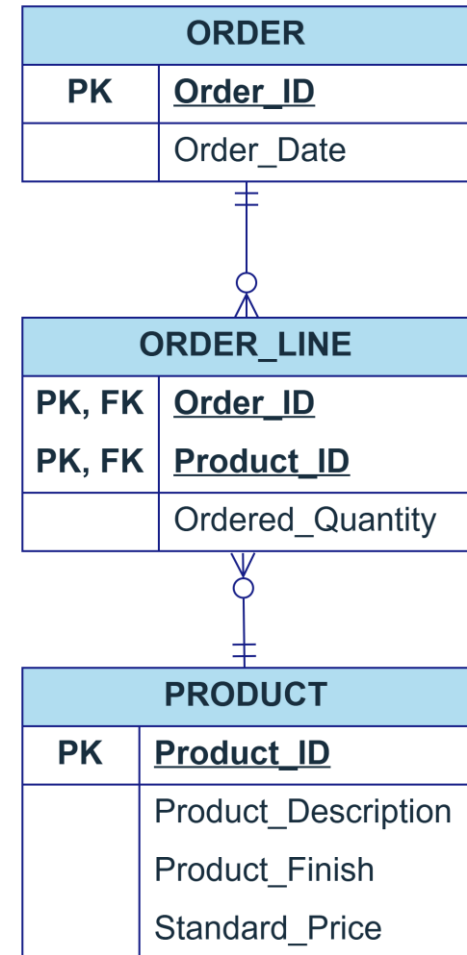
<u>Order_ID</u>	Order_Date
101	11/1/2024
102	11/3/2024
103	11/5/2024

ORDER_LINE

<u>Order_ID</u>	<u>Product_ID</u>	Ordered_Quantity
1001	P101	2
1001	P103	1
1002	P102	3
1002	P103	1
1003	P101	1

PRODUCT

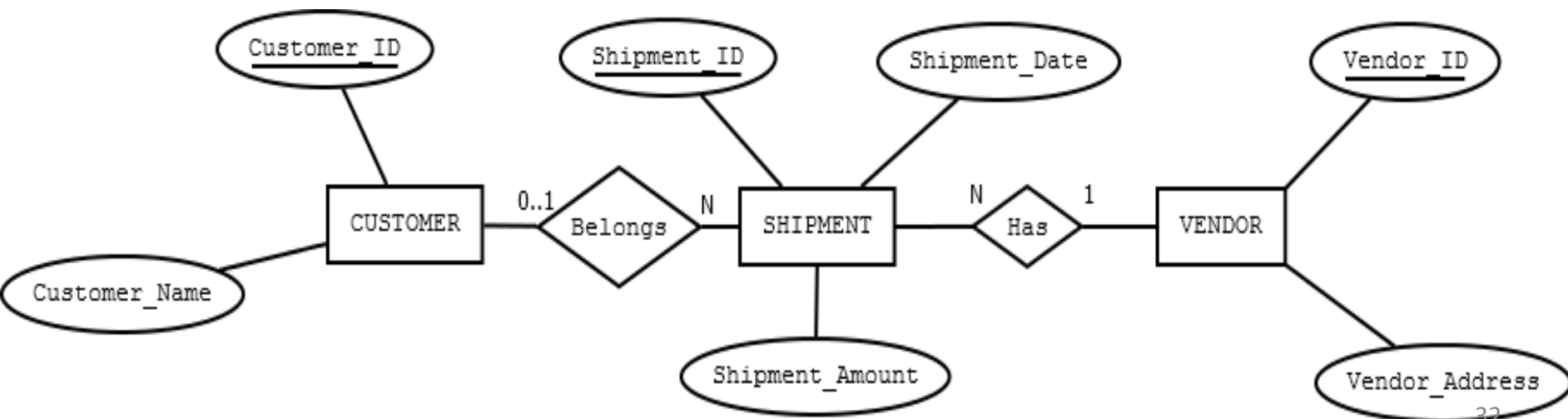
<u>Product_ID</u>	Product_Description	Product_Finish	Standard_Price
P101	Laptop	Silver	1200
P102	Smartphone	Black	799.99
P103	Headphones	White	199.99
P104	Tablet	Space Grey	499.99



Mapping Associative Entities

b. Identifier is assigned

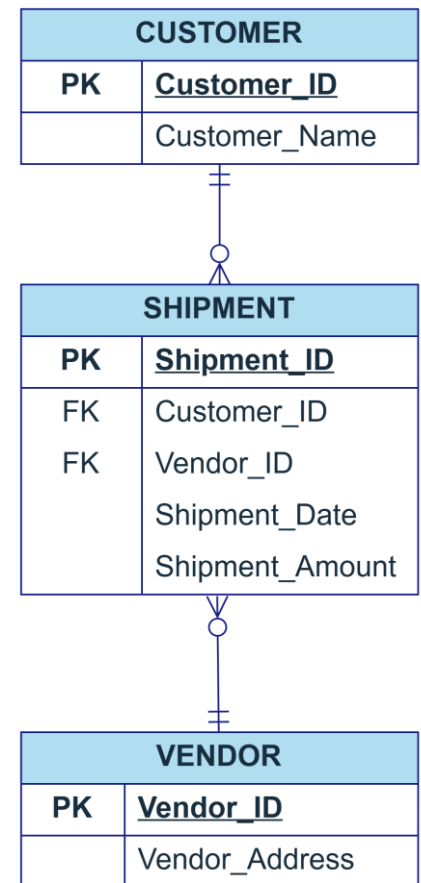
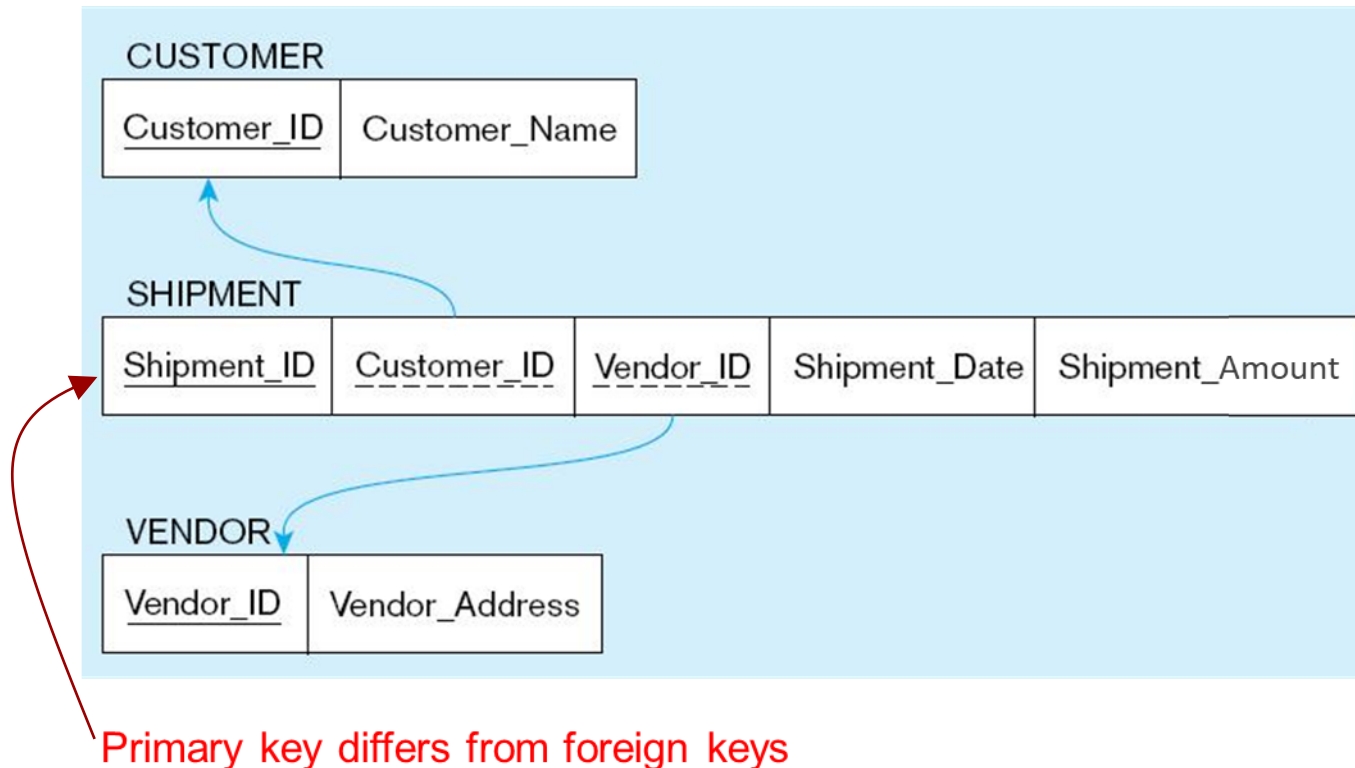
- It is natural and familiar to end-users.
- Default identifier may not be unique. So the primary key may be a composition of default identifier and foreign keys.
- Example : Mapping an associative entity, with identifier assigned: SHIPMENT associative entity



Mapping Associative Entities

Example : Mapping an associative entity, with identifier assigned

– Three resulting relations



Mapping Associative Entities

Example : Mapping an associative entity, with identifier assigned

– Three resulting relations

CUSTOMER

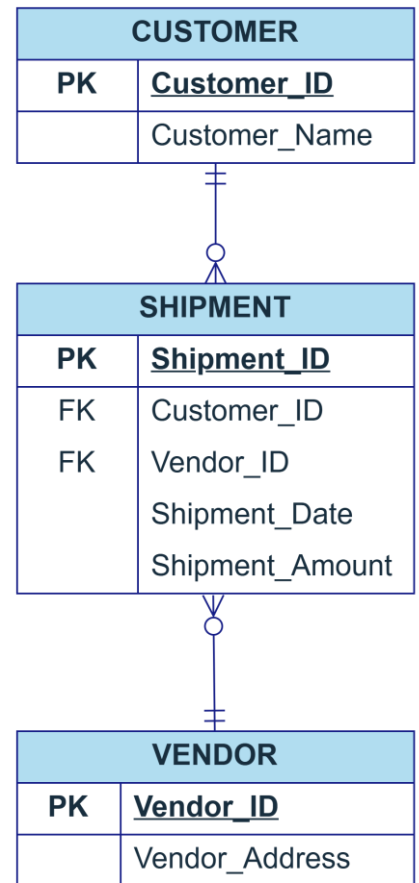
<u>Customer_ID</u>	CustomerName
1	John Doe
2	Jane Smith
3	Michael Brown
4	Emily Johnson

VENDOR

<u>Vendor_ID</u>	Vendor_Address
101	123 Vendor St, NY
102	456 Supplier Ave, LA
103	789 Wholesale Blvd, SF

SHIPMENT

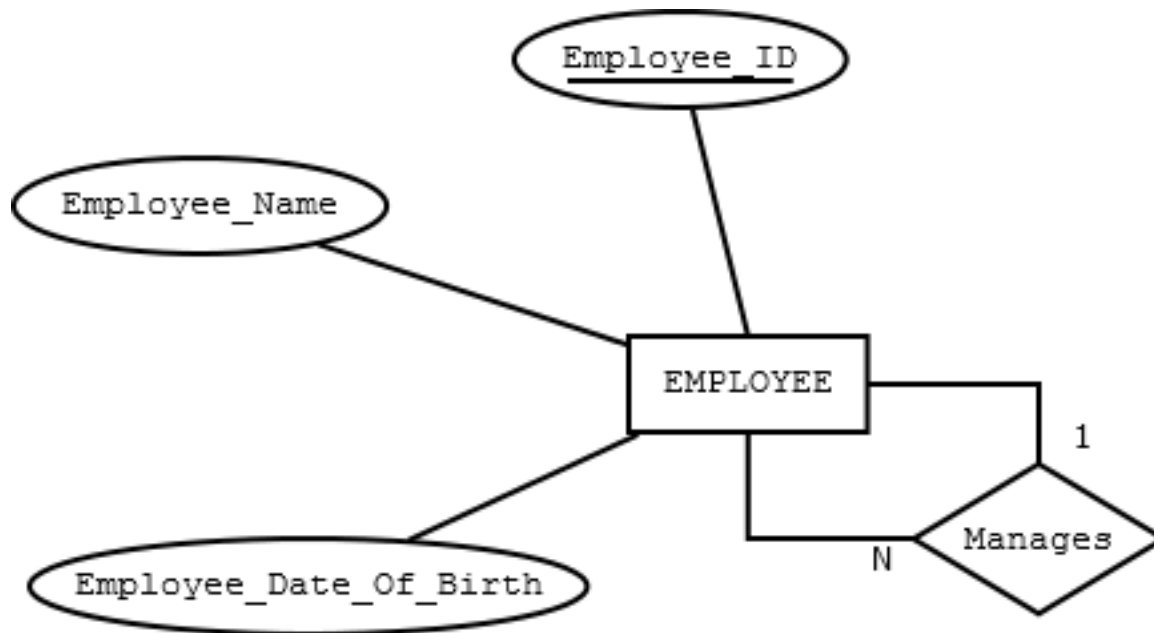
<u>Shipment_ID</u>	Customer_ID	Vendor_ID	Shipment_Date	Shipment_Amount
1001	1	101	9/1/2024	150
1002	1	102	9/3/2024	200
1003	3	103	9/5/2024	50
1004	2	102	9/7/2024	100



Mapping Unary Relationships

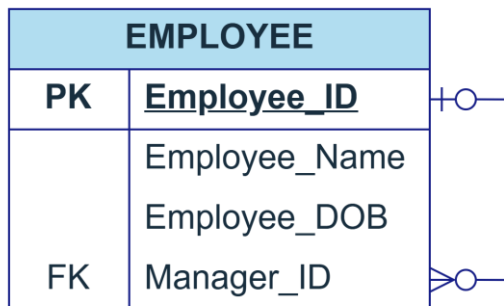
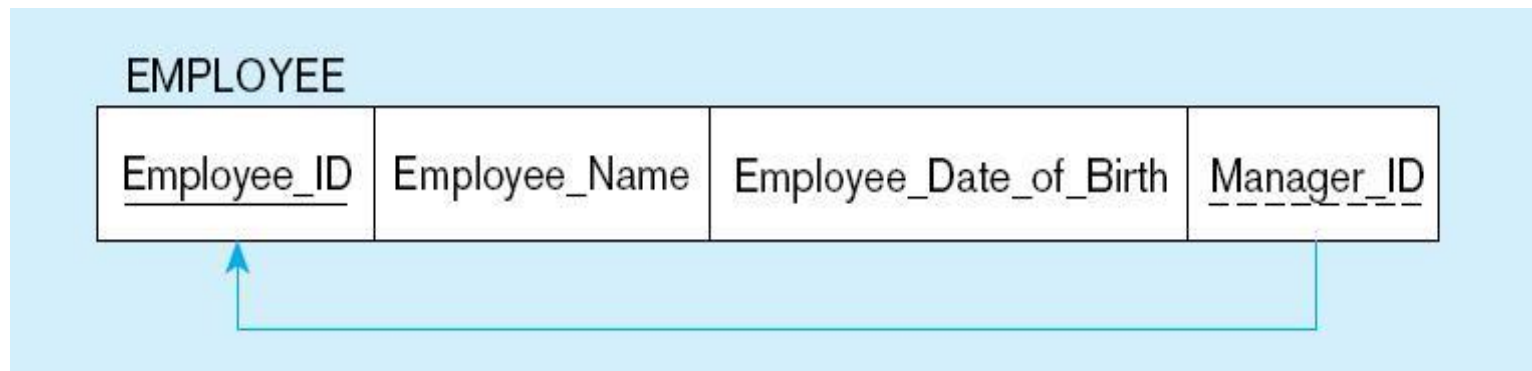
A. Mapping a unary 1:1 or 1:N relationship:

- Recursive foreign key in the same table.
- Example: Mapping a unary 1:N relationship
 - EMPLOYEE entity with unary relationship



Mapping Unary Relationships

- Example: Mapping a unary 1:N relationship
 - EMPLOYEE relation with recursive foreign key



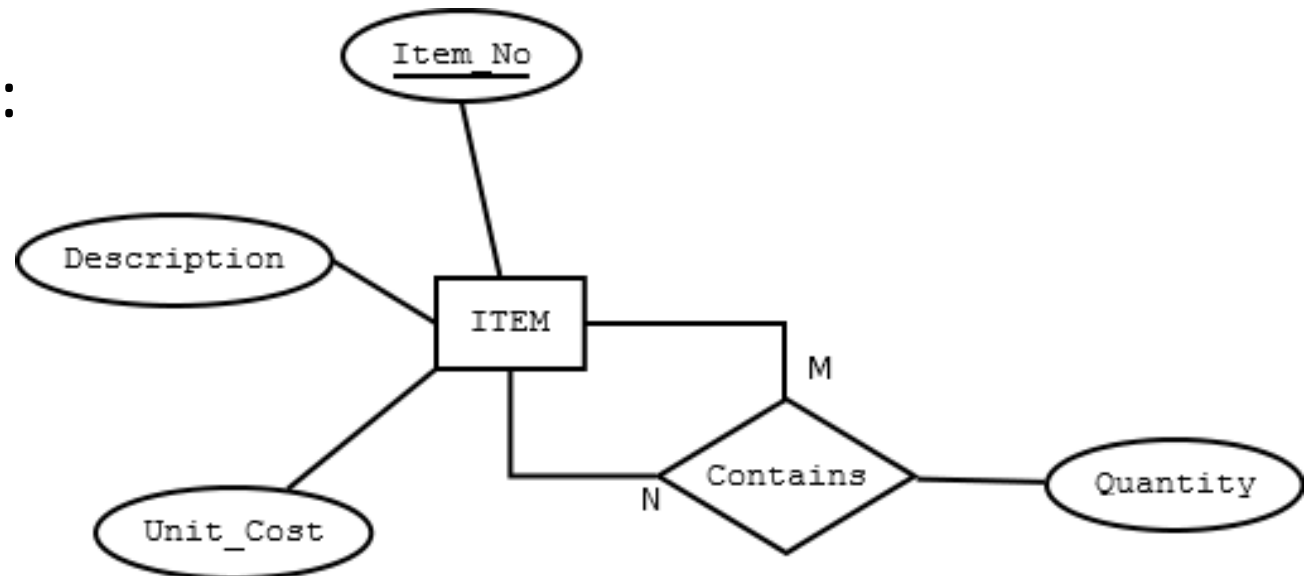
EMPLOYEE

<u>Employee_ID</u>	Employee_Name	Employee_DOB	Manager_ID
1	Alice Johnson	5/15/1980	NULL
2	Bob Smith	7/20/1985	1
3	Carol Miller	2/10/1990	1
4	David Brown	8/25/1992	2
5	Eve White	11/30/1995	2

Mapping Unary Relationships

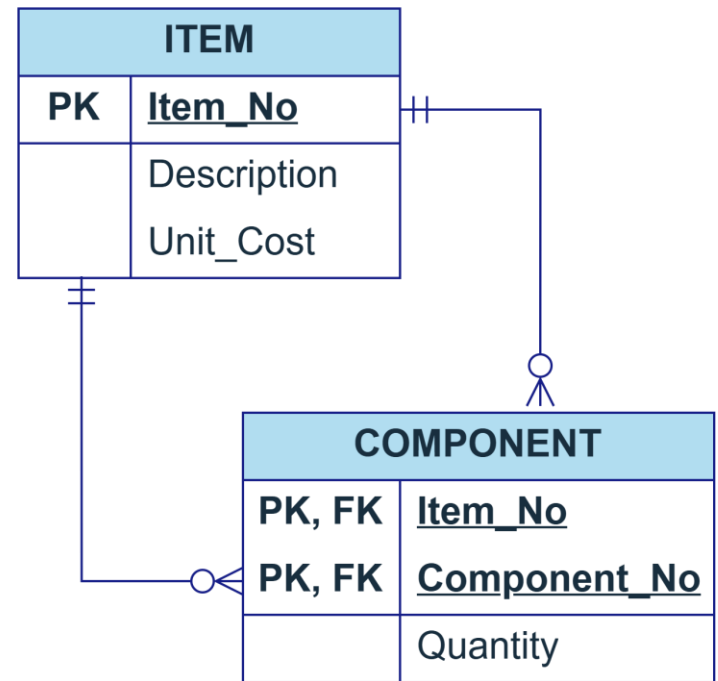
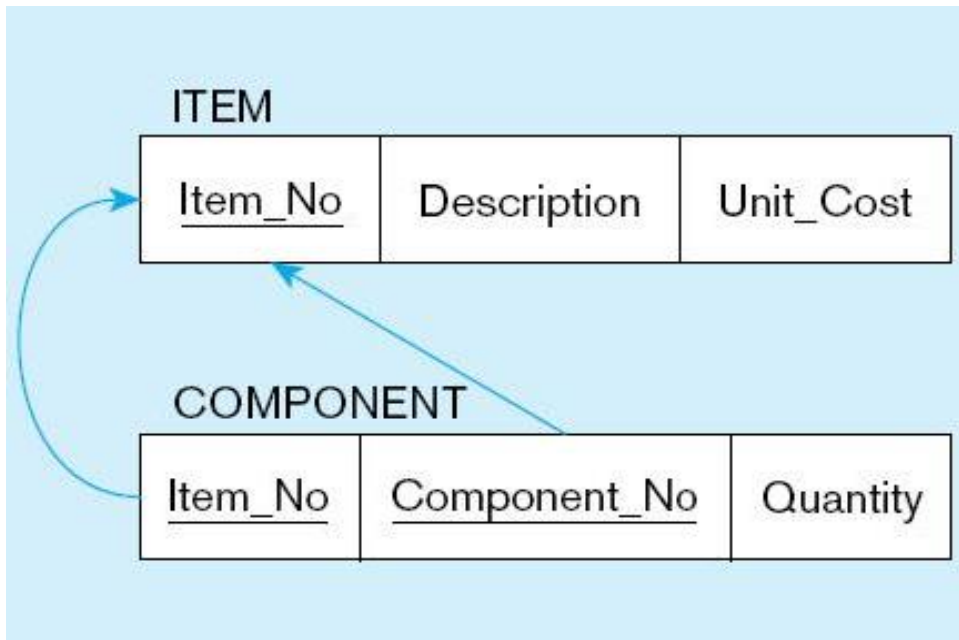
B. Mapping a unary Many-to-Many relationship:

- One for the entity type.
- One for an associative relation in which the primary key has two attributes, both taken from the primary key of the entity.
- Example:



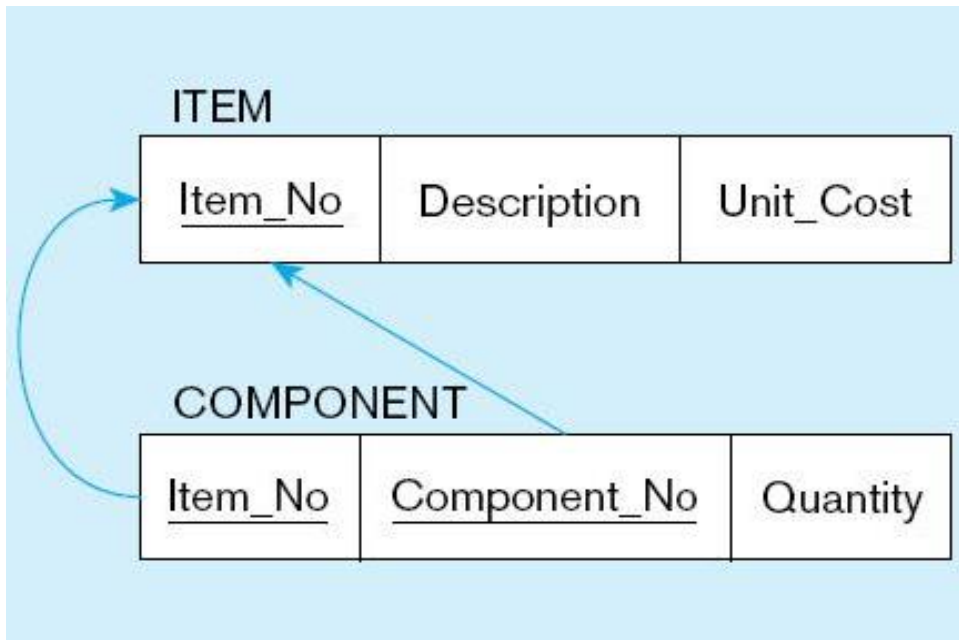
Mapping Unary Relationships

- Example: Mapping a unary M:N relationship
 - ITEM and COMPONENT relations



Mapping Unary Relationships

- Example: Mapping a unary M:N relationship
 - ITEM and COMPONENT relations



ITEM

<u>Item_No</u>	Description	Unit_Cost
1	Bicycle	300
2	Wheel	50
3	Seat	20
4	Handlebar	15

COMPONENT

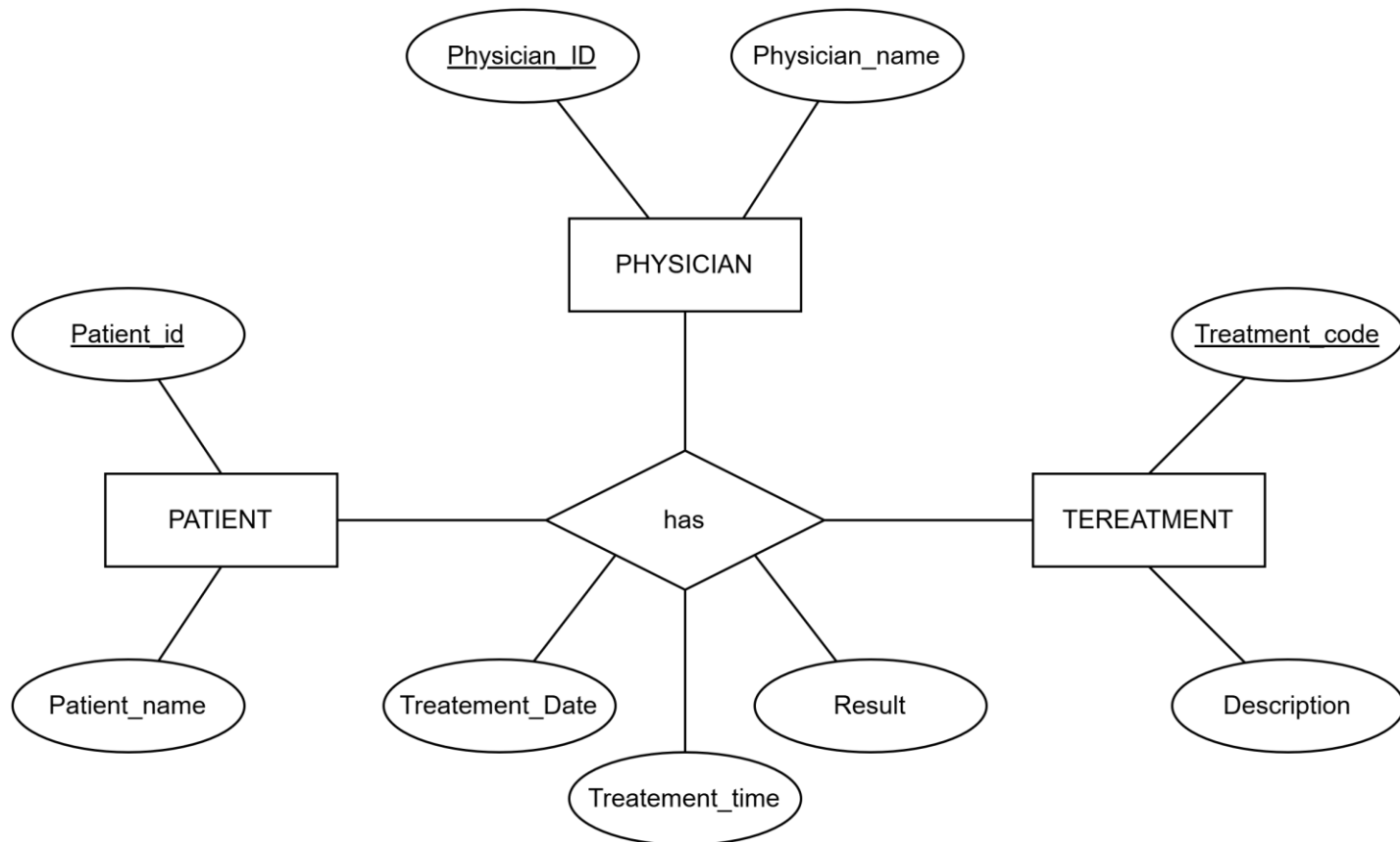
<u>Item_No</u>	<u>Component_No</u>	Quantity
1	2	2
1	3	1
1	4	1
2	4	2

Mapping Ternary (and n-ary) Relationships

- Mapping Ternary (and n-ary) Relationships
 - One relation for each entity and one for the associative entity.
 - Associative entity has foreign keys of each entity in the relationship.

Mapping Ternary (and n-ary) Relationships

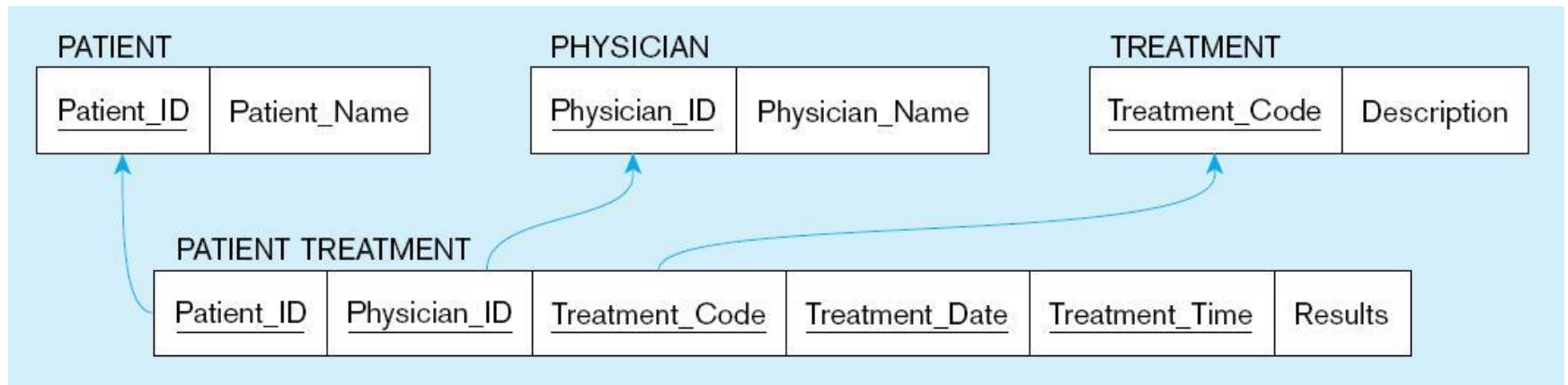
- Example: Mapping a Ternary relationship
 - PATIENT TREATMENT Ternary relationship with associative entity



Mapping Ternary (and n-ary) Relationships

- Example: Mapping a Ternary relationship
- Mapping the ternary relationship PATIENT TREATMENT

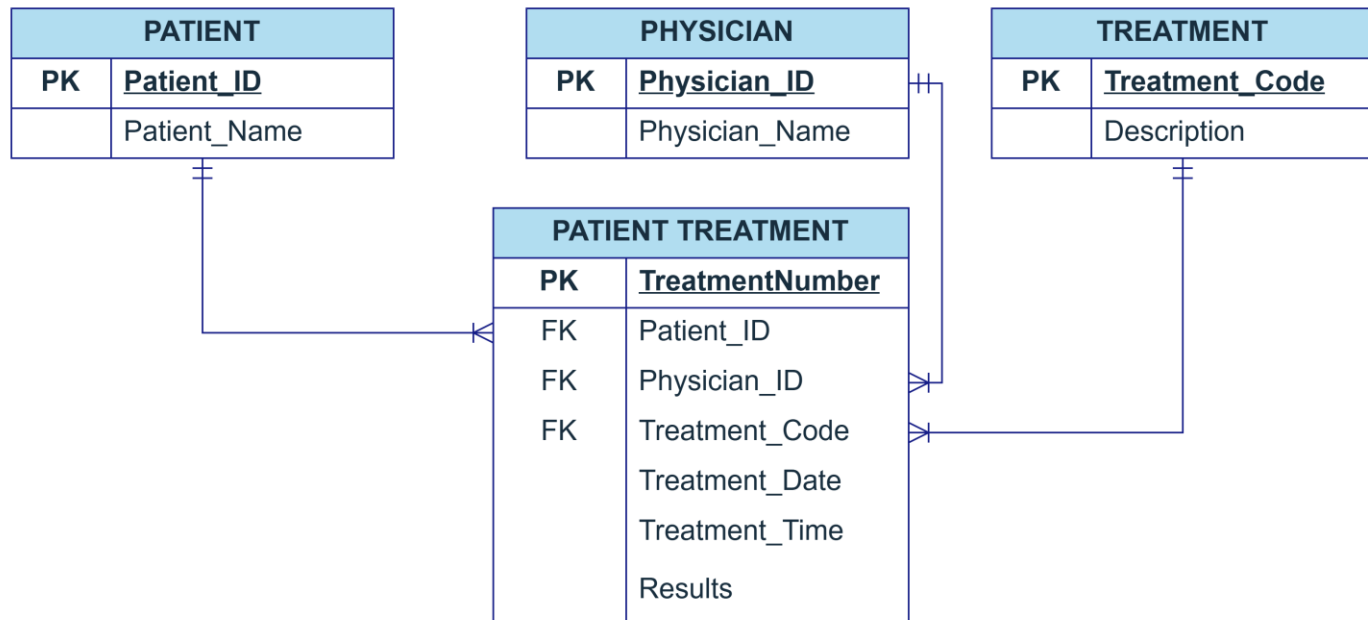
Remember that the primary key MUST be unique. This is why treatment **date** and **time** are included in the composite primary key. But this makes a very cumbersome key. It would be better to create a **surrogate key** like **TreatmentNumber**.



Mapping Ternary (and n-ary) Relationships

- Example: Mapping a Ternary relationship
- Mapping the ternary relationship PATIENT TREATMENT

Remember that the primary key MUST be unique. This is why treatment **date** and **time** are included in the composite primary key. But this makes a very cumbersome key. It would be better to create a **surrogate key** like **TreatmentNumber**.



Mapping Ternary (and n-ary) Relationships

- Example: Mapping a Ternary relationship
 - Mapping the ternary relationship PATIENT TREATMENT

Remember that the primary key MUST be unique. This is why treatment **date** and **time** are included in the composite primary key. But this makes a very cumbersome key. It would be better to create a **surrogate key** like **TreatmentNumber**.

Example: PATIENT TREATMENT TABLE

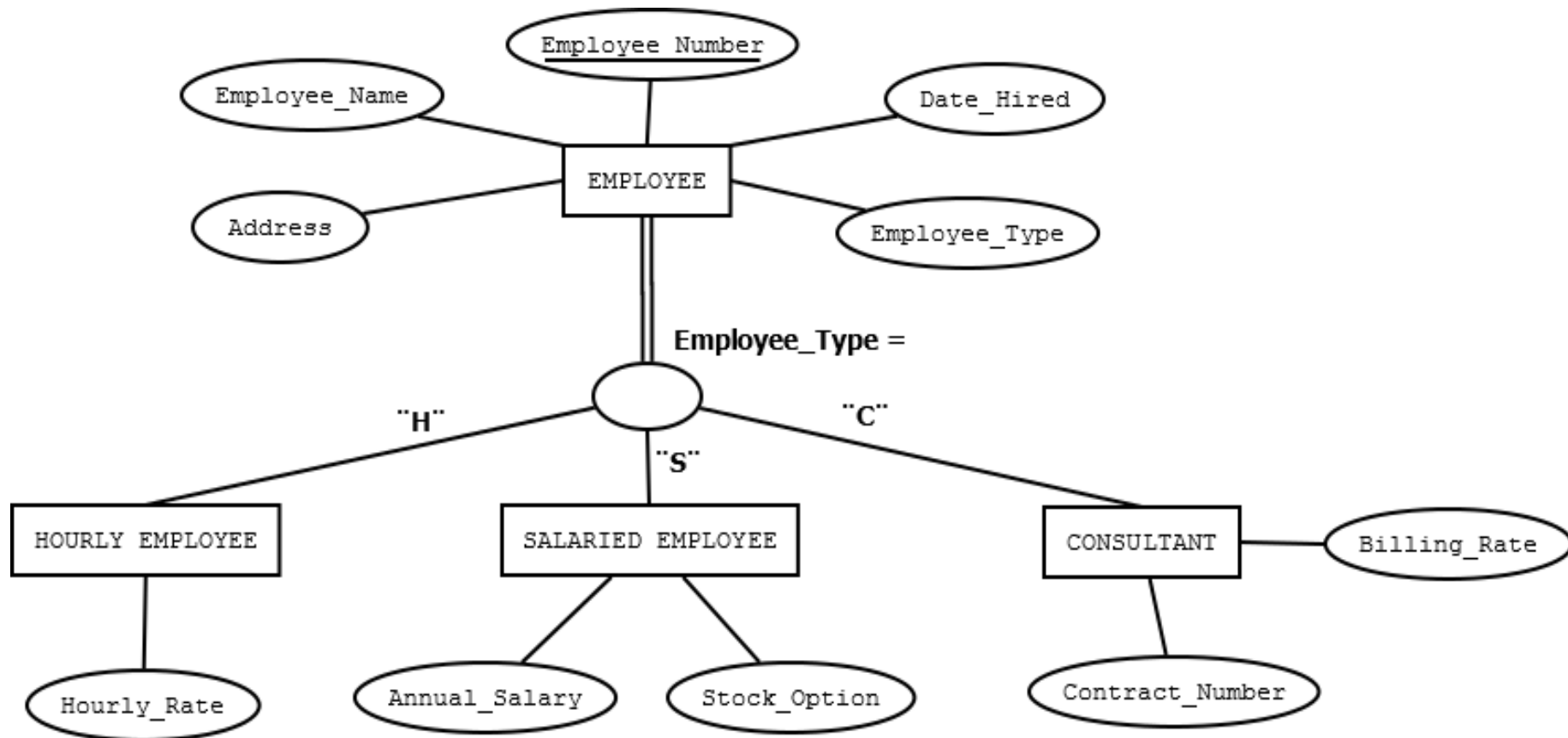
<u>TreatmentNumber</u>	Patient_ID	Physician_ID	Treatment_Code	Treatment_Date	Treatment_Time	Results
1001	1	101	T001	2024-05-01	11:00:00	Successful surgery
1002	1	101	T001	2024-07-01	9:00:00	Recovery Progressing
1003	2	101	T002	2024-09-01	14:30:00	Successful recovery

Mapping Supertype/Subtype Relationships

- Mapping Supertype/Subtype Relationships
 - One table for supertype and separate tables for each subtype.
 - Attributes of supertype into supertype table
 - Subtype attributes go into each subtype; primary key of supertype relation also becomes primary key of subtype relation
 - 1:1 relationship established between supertype and each subtype, with supertype as primary table.

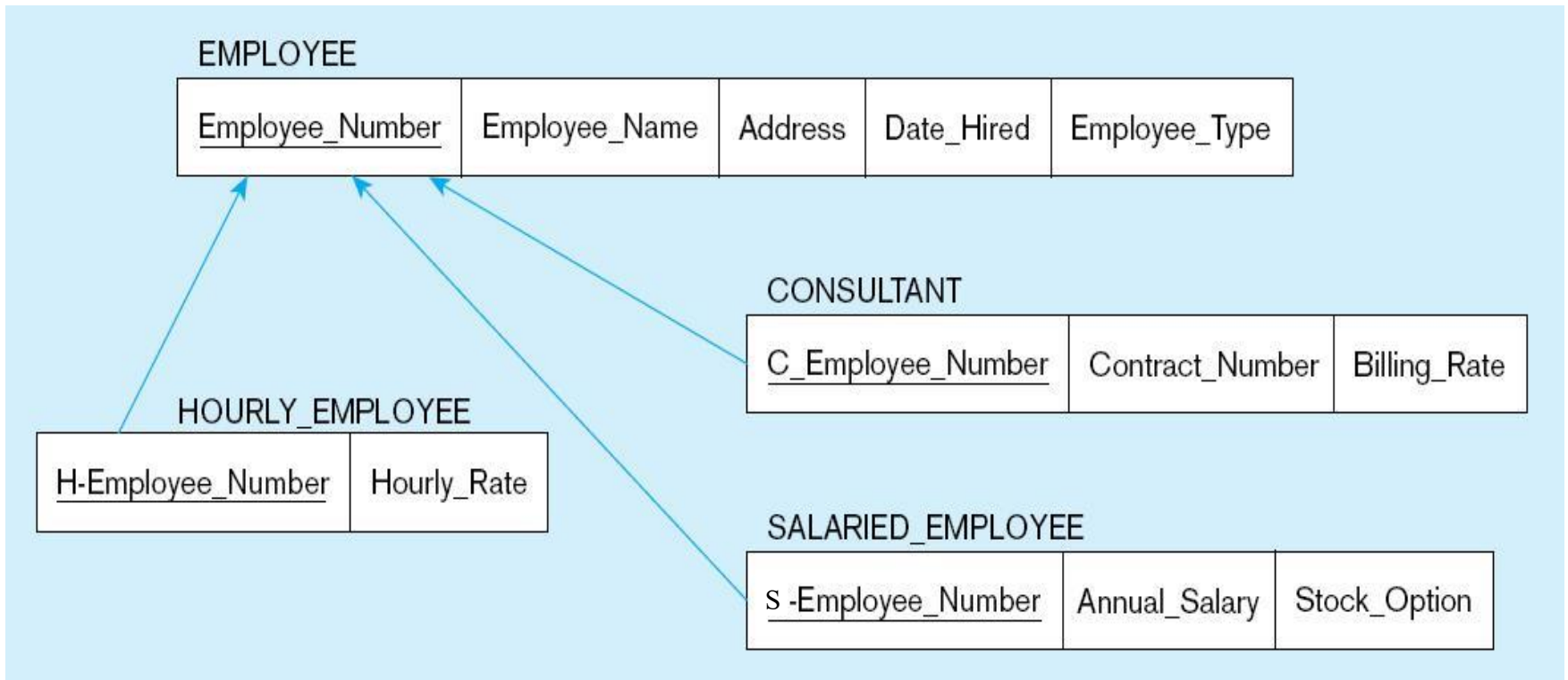
Mapping Supertype/Subtype Relationships

- Example: Mapping supertype/subtype relationships



Mapping Supertype/Subtype Relationships

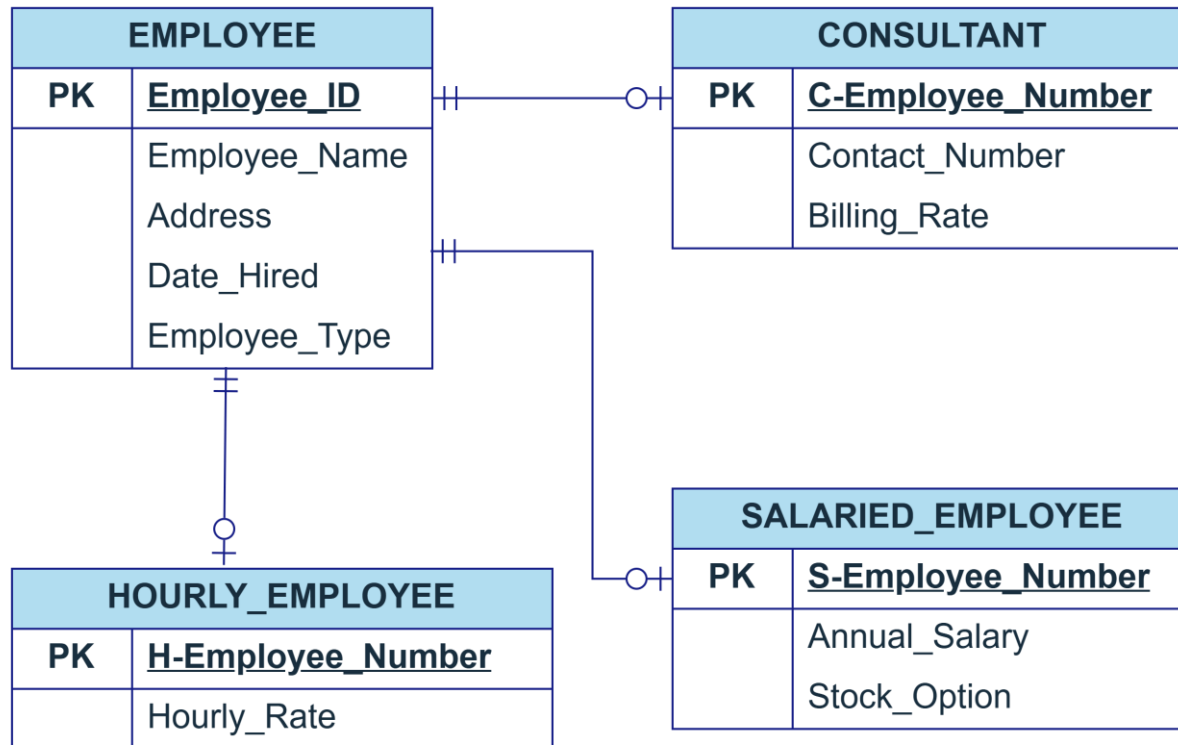
- Example: Mapping supertype/subtype relationships



These are implemented as one-to-one relationships.

Mapping Supertype/Subtype Relationships

- Example: Mapping supertype/subtype relationships



These are implemented as one-to-one relationships.

Mapping Supertype/Subtype Relationships

– Example: Mapping supertype/subtype relationships

EMPLOYEE

<u>Employee Number</u>	Employee_Name	Address	Date_Hired	Employee_Type
101	John Doe	123 Elm St, NY	1/15/2023	Hourly
102	Jane Smith	456 Oak St, LA	6/10/2022	Consultant
103	Alice Johnson	789 Pine St, SF	3/21/2021	Salaried

HOURLY_EMPLOYEE

<u>H-Employee Number</u>	Hourly_Rate
101	25

SALARIED_EMPLOYEE

<u>S-Employee Number</u>	Annual_Salary	Stock_Option
103	60000	5000

CONSULTANT

<u>C-Employee Number</u>	Contact_Number	Billing_Rate
102	555-1234	75

These are implemented as one-to-one relationships.

Case study

- A case study used as the example

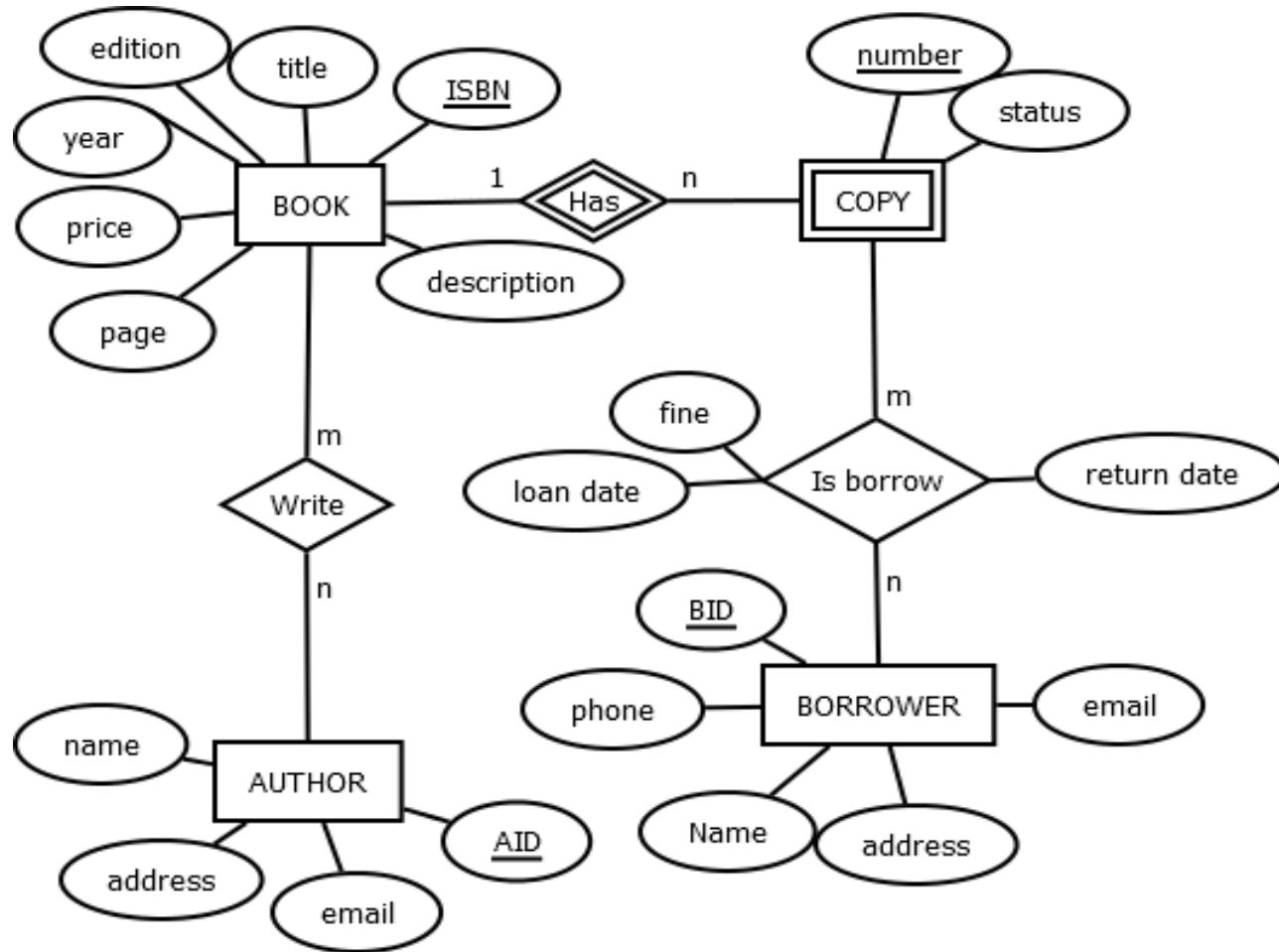
This case study describes a simple Library Management System. Assume the requirements for the system were written by your client exactly in these terms:

"The system will manage author's and loaner's information, and keep track of books loaned. The borrower's information include name, address, e-mail, and phone. The author's information include name, address and e-mail.

New books, authors and clients are entered into the system. When a client checks out a book, the system will register the date the book was loaned and calculate the days the book can be loaned. It will also calculate the date the book is due to be returned. If the borrower returns the book late, he must pay a fine based on the number of days overdue."

Case study

- Conceptual Design With Entity Relationship Model for a simple Library Management System



- Convert this diagram to the logical relational data model

Case Study

Convert this diagram to the relational data model

1. BOOK(ISBN, title, edition, year, price, page)
2. COPY(number, ISBN(FK), status)
3. AUTHOR(AID, name, address, email)
4. BORROWER(BID, name, address, phone, email)
5. AUTHOR_LIST(ISBN(FK), AID(FK))
6. LOAN(number(FK), ISBN(FK), BID(FK),
loan_date, return_date, fine)

Case Study

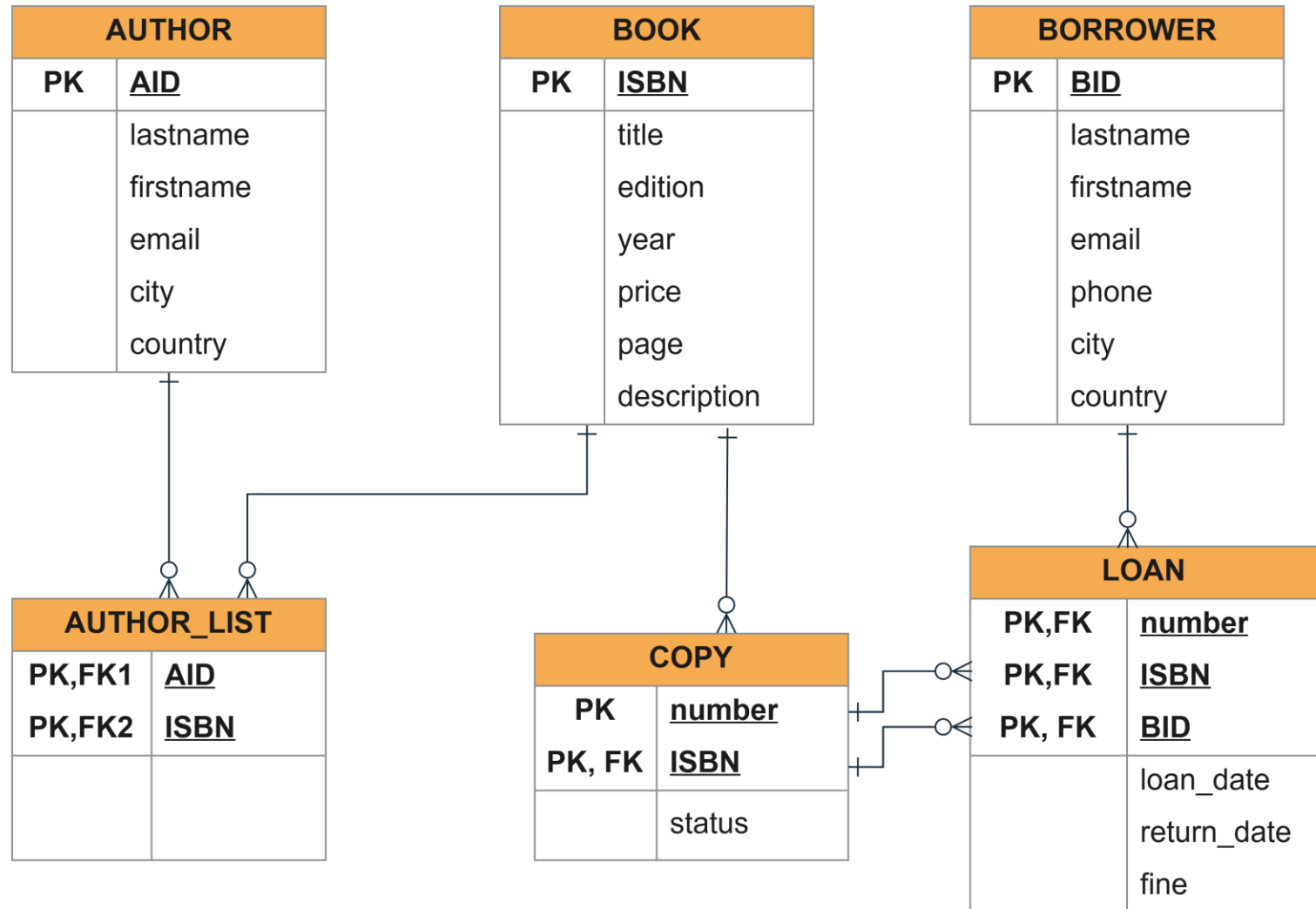


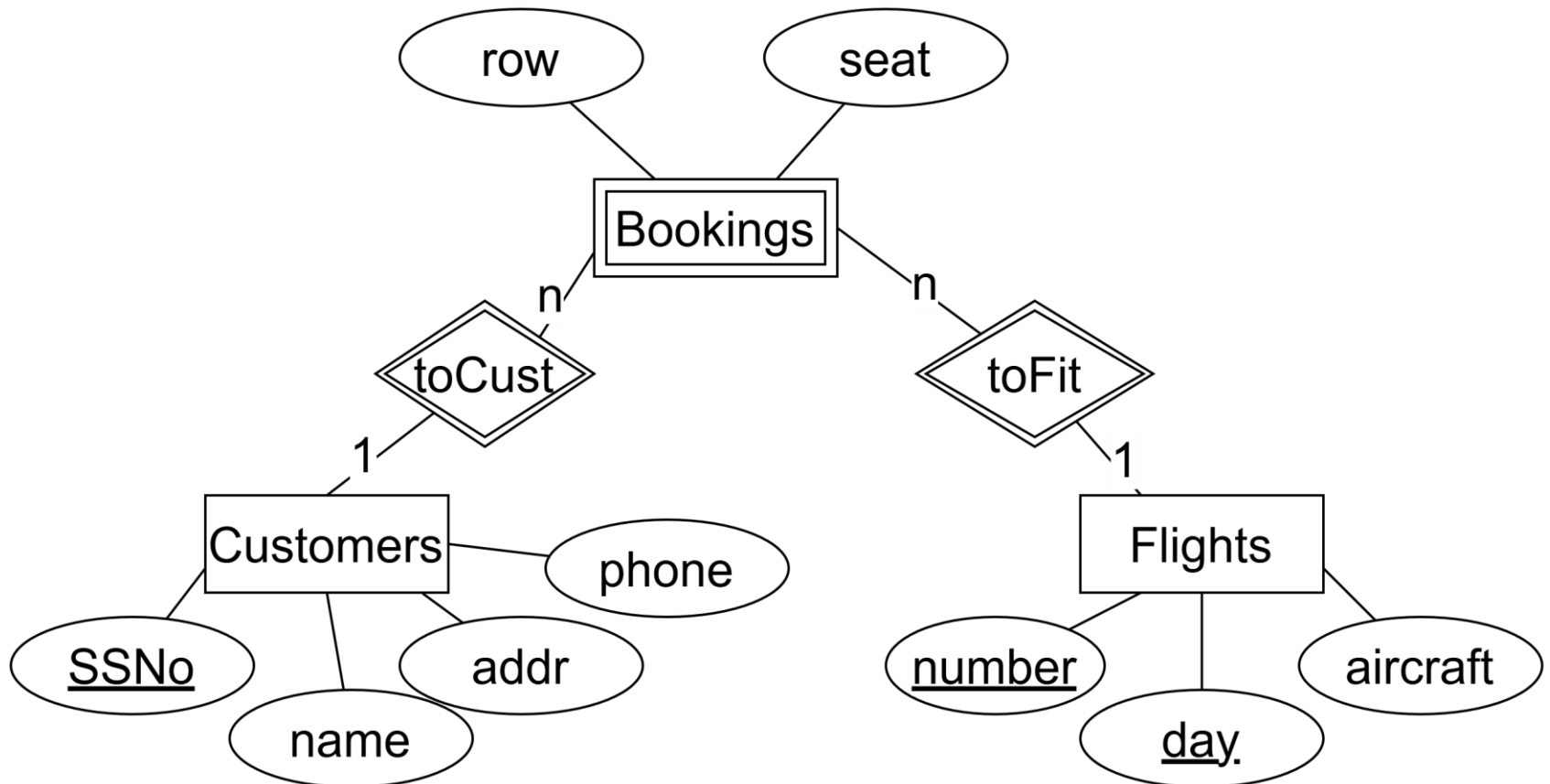
Figure – the logical model

Summary

No	Topic	Summary
1	Map regular entities	The mapping between regular entities into relational model is straight forward.
2	Map weak entities	The weak entity set becomes a separate relation with a foreign key taken from the superior entity.
3	Map binary relationships	Additional table may be created depends on the relationship type.
4	Map associative entities	A separate table is created for the associative entity.
5	Map unary relationships	Additional table will be created for M:N relationship. In case of 1:1 relationship we use the recursive foreign key as the primary key in the same relation.
6	Map ternary relationships	We need to create one relation for each entity and one for the associative entity.
7	Map supertype/subtype relationships	We need to create one relation for supertype and for each subtype.

Assignment

- Exercise 1: Convert the E-R diagram to a relational database schema.



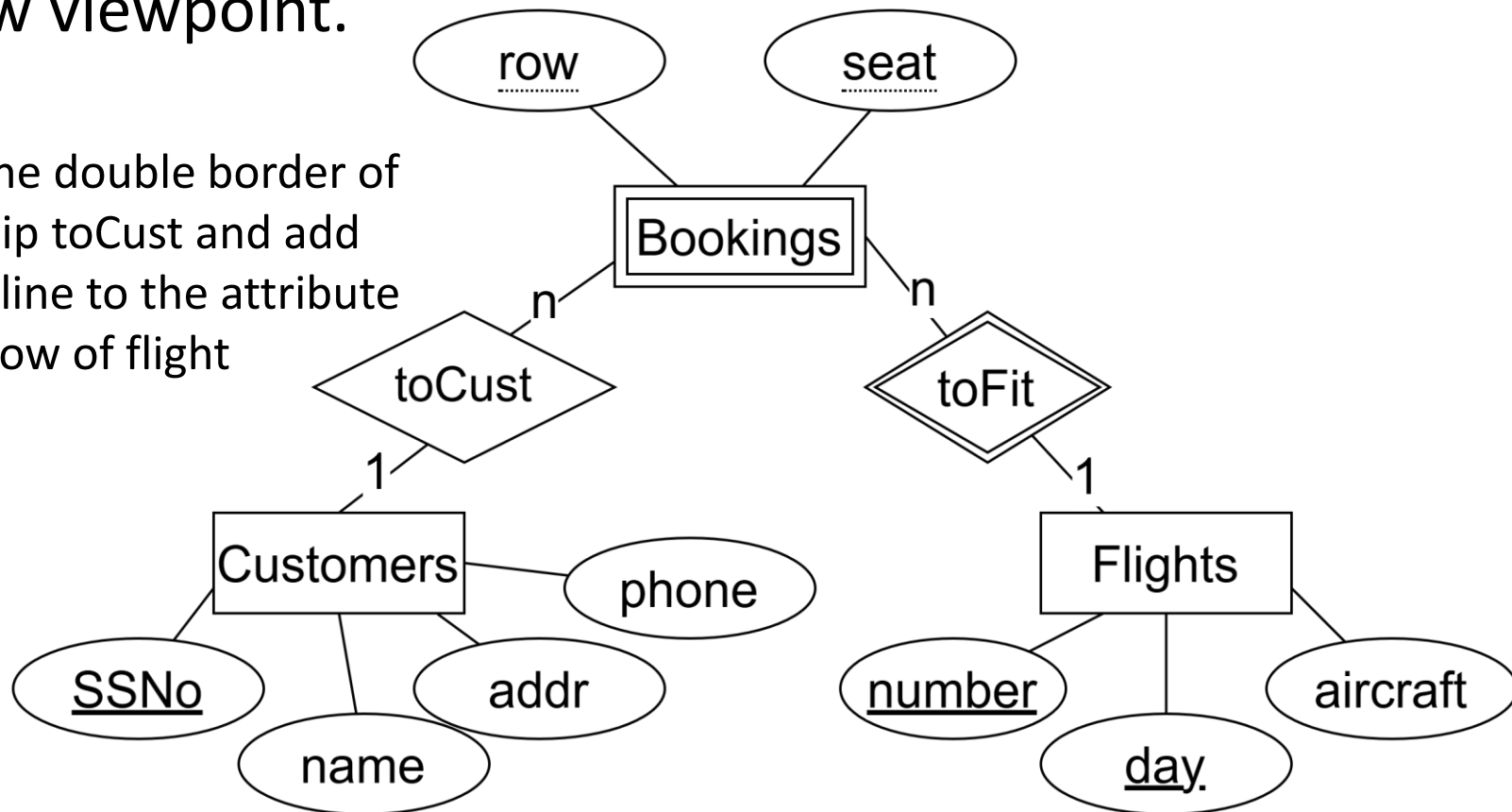
Assignment

- Exercise 2:
 - There is another E-R diagram that could describe the weak entity set Bookings in E-R diagram in Exercise 1. Notice that a booking can be identified uniquely by the flight number, day of the flight, the row, and the seat; the customer is not then necessary to help identify the booking.
 - (a) Revise the E-R diagram of Exercise 1 to reflect this new viewpoint.
 - (b) Convert your diagram from (a) into tables. Do you get the same database schema as in Exercise 1?

Answer

- (a) Revise the E-R diagram of Exercise 1 to reflect this new viewpoint.

Remove the double border of relationship toCust and add dot underline to the attribute seat and row of flight



Assignment

Exercise 3:

- The E-R diagram below represents ships. Ships are said to be sisters if they were designed from the same plans. Convert this diagram to a relational database schema.

- A). 1:1 or 1:N
- B). M:N

