DATABASE CONCEPTS &
ER MODEL

Learning Goals

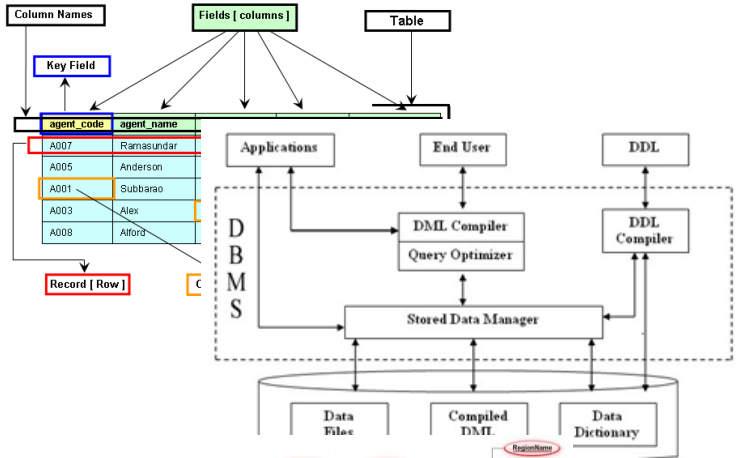
By the end of this lecture students should be able to:

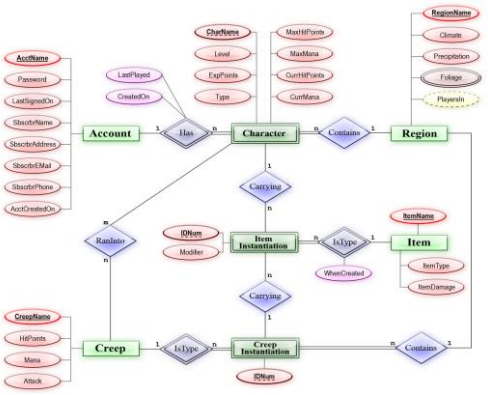
Understand an overview of the basic RDBMS Concepts

Understand an insight into the architecture and components of a Database System.

Describe how entities, attributes and relationships are used to model data;

Converting ER Model to relational schema





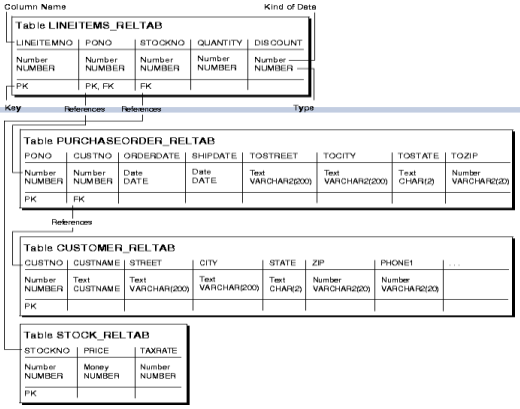


Table of contents

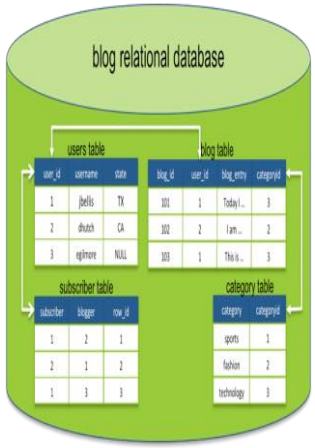
Section 1

SQL OVERVIEW

Structured

Query

Language



- **Which is a computer language for:**
  - ✓ storing,
  - ✓ manipulating and
  - ✓ retrieving data stored in relational database.
- SQL is the standard language for Relation Database System, like **MySQL**, **MS Access**, **Oracle**, **Sybase**, **Informix**, **Postgres** and **SQL Server** use SQL as standard database language.
- SQL is an ANSI (American National Standards Institute) standard.

SQL Process

SQL Query

Query Language Processor

Parser + Opimizer

DBMS Engine

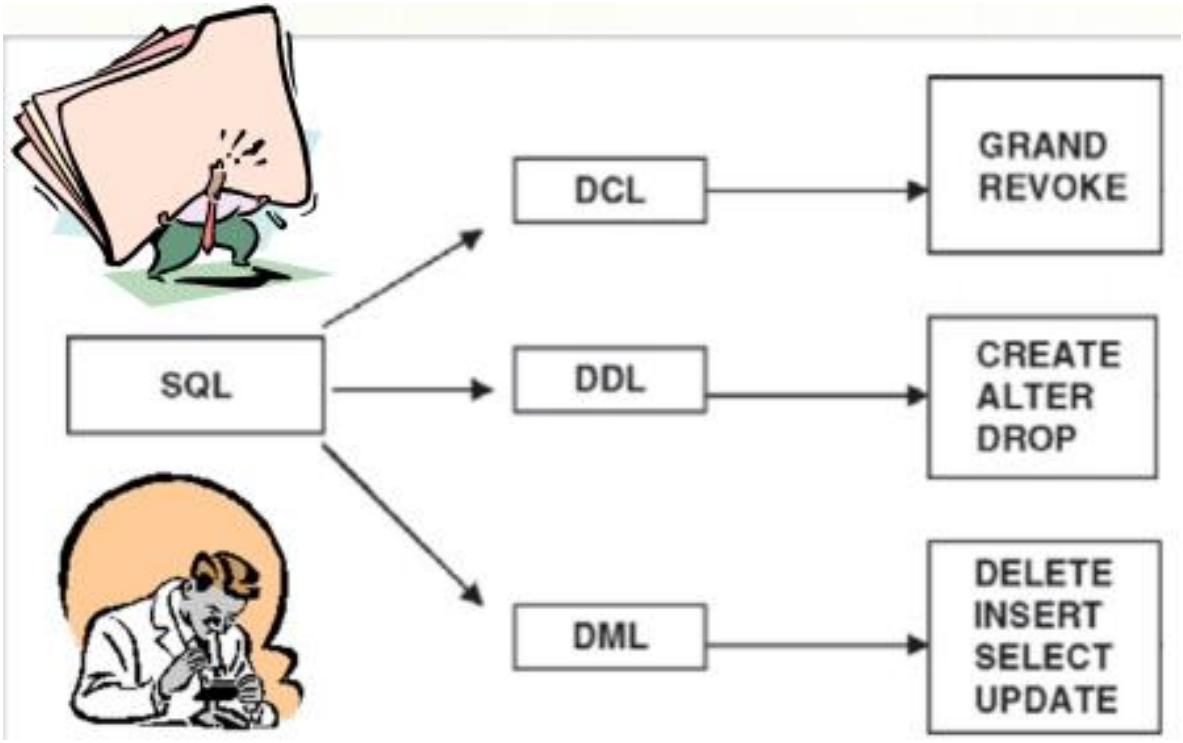
File Manager
+
Transaction manager

Physical Database

SQL Commands

SQL consists of **three components:**

- Data Definition Language (DDL)
- Data Manipulation Language (DML) and
- Data Control Language (DCL)

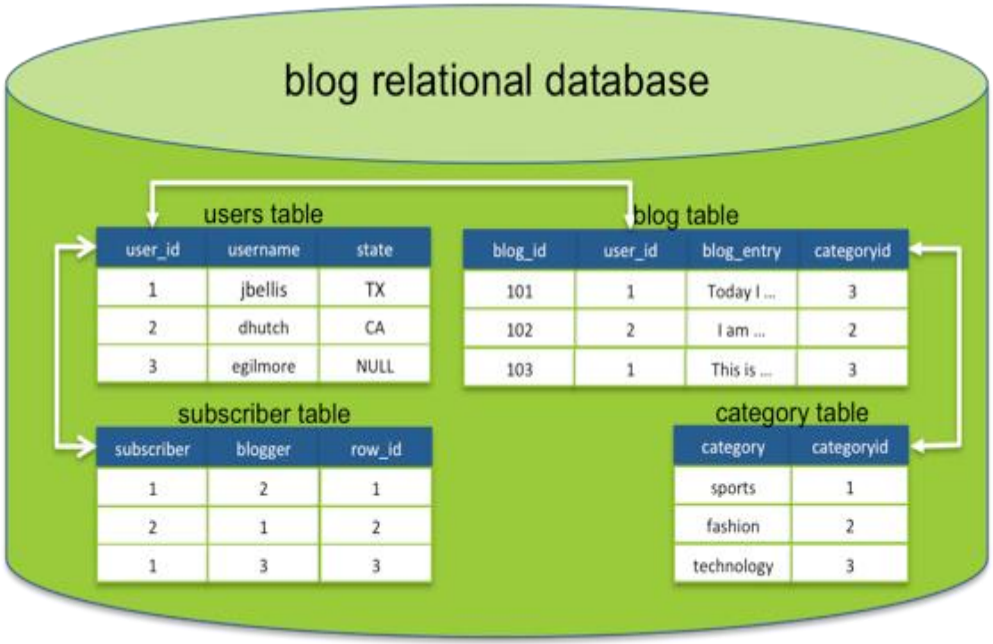


Section 2

THE RELATIONAL DATABASE

Relational Database Concepts (1/3)

- "A DBMS that manages data as collection of **tables** in which all data relationships are represented by common values in related tables."
- "A DBMS that follows all the twelve rules of CODD is called RDBMS"



Relational Database Concepts (2/3)

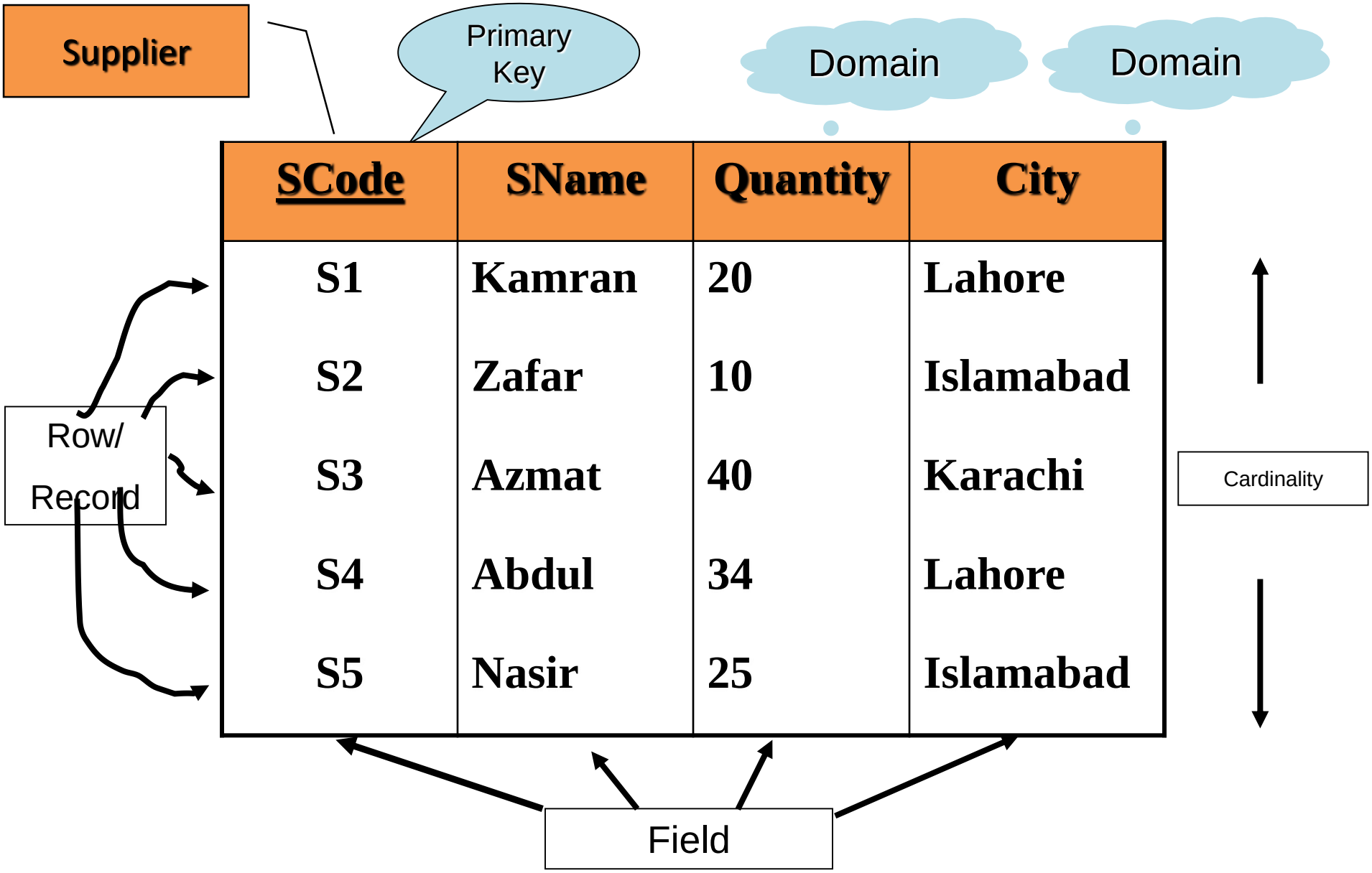
Resource Software Solution

Table

Field

Record

CD_ID	Title	Artist	Genre
1	The Wall	Pink Floyd	Rock
2	Blue Train	John Coltrane	Jazz
3	Requiem	W.A. Mozart	Classical



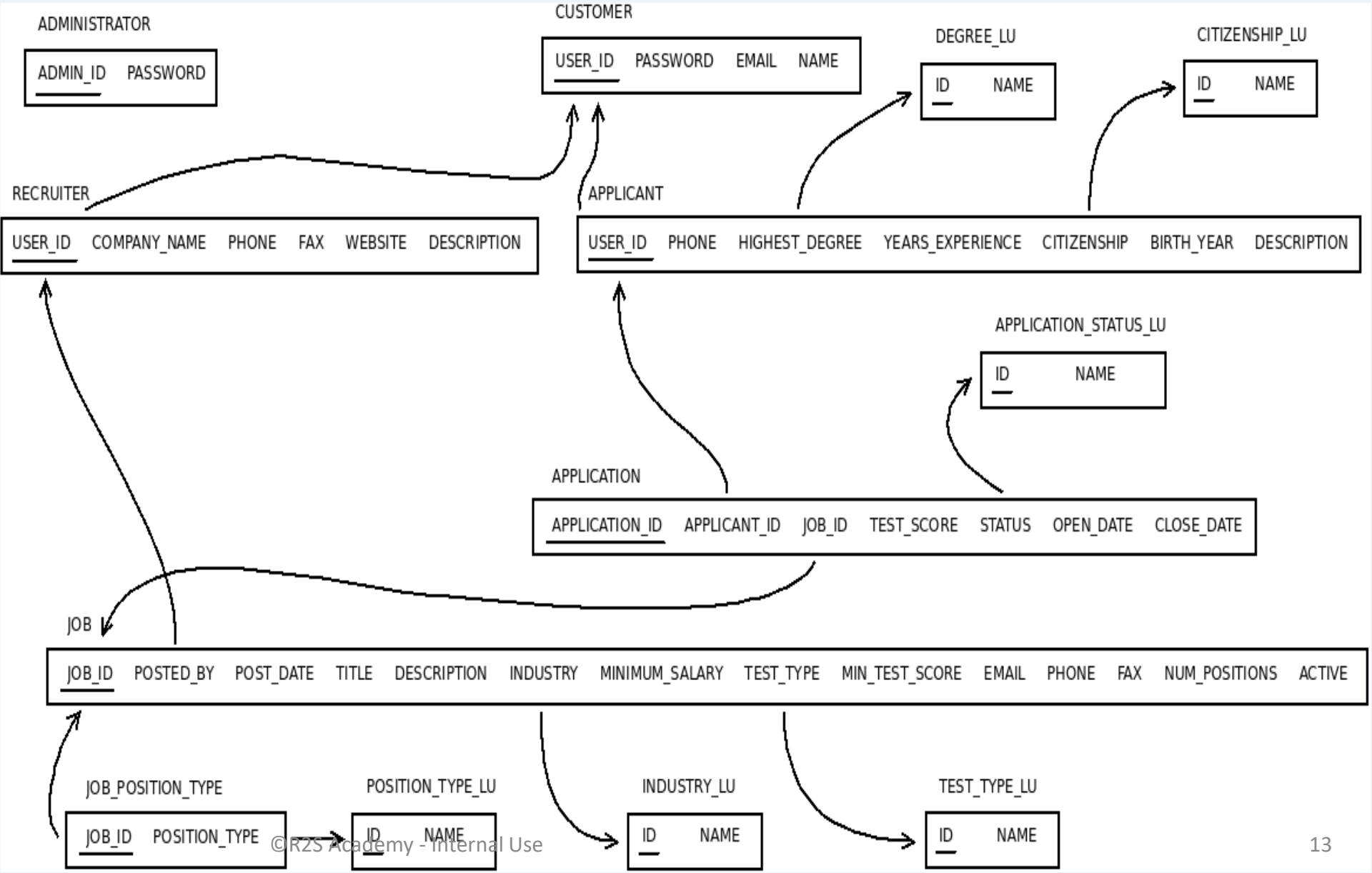
Schema (1/2)

- The name of a relation and the set of attributes for a relation is called a **schema**.
  - Example: the schema for previous slide is

Supplier (SCode, SName, Quantity, City)

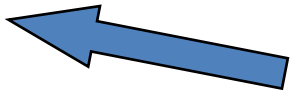
- **Relation schema** = name(attributes) + other structure info., e.g., keys, other constraints.
- Order of attributes is arbitrary, but in practice we need to assume the (*standard*) order given in the relation schema.
- **Relational database schema** = collection of relation schemas.

Schema (2/2)



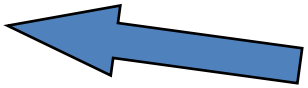
Schema versus Instance

Student(studno,name,address)
Course(courseno,lecturer)



Schema

Student(123,Bloggs,Woolton)
(321,Jones,Owens)



Instance

sid	Name	Login	age	GPA
53666	Jones	Jones@ca	18	3.4
53444	smith	Smith@ecs	18	3.2
53777	Blake	Blake@aa	19	3.8

➔ Cardinality = 3, arity = 5 , all rows distinct

➔ Do all values in each column of a relation instance have to be distinct?

What is RDBMS?

- **RDBMS** stands for:

  Relational **D**atabase **M**anagement **S**ystem

- **RDBMS** is the basis for SQL, and for all modern database systems like:
  - ✓ MS SQL Server,
  - ✓ IBM DB2,
  - ✓ Oracle,
  - ✓ MySQL,
  - ✓ and Microsoft Access.
- A Relational database management system (RDBMS) is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd.

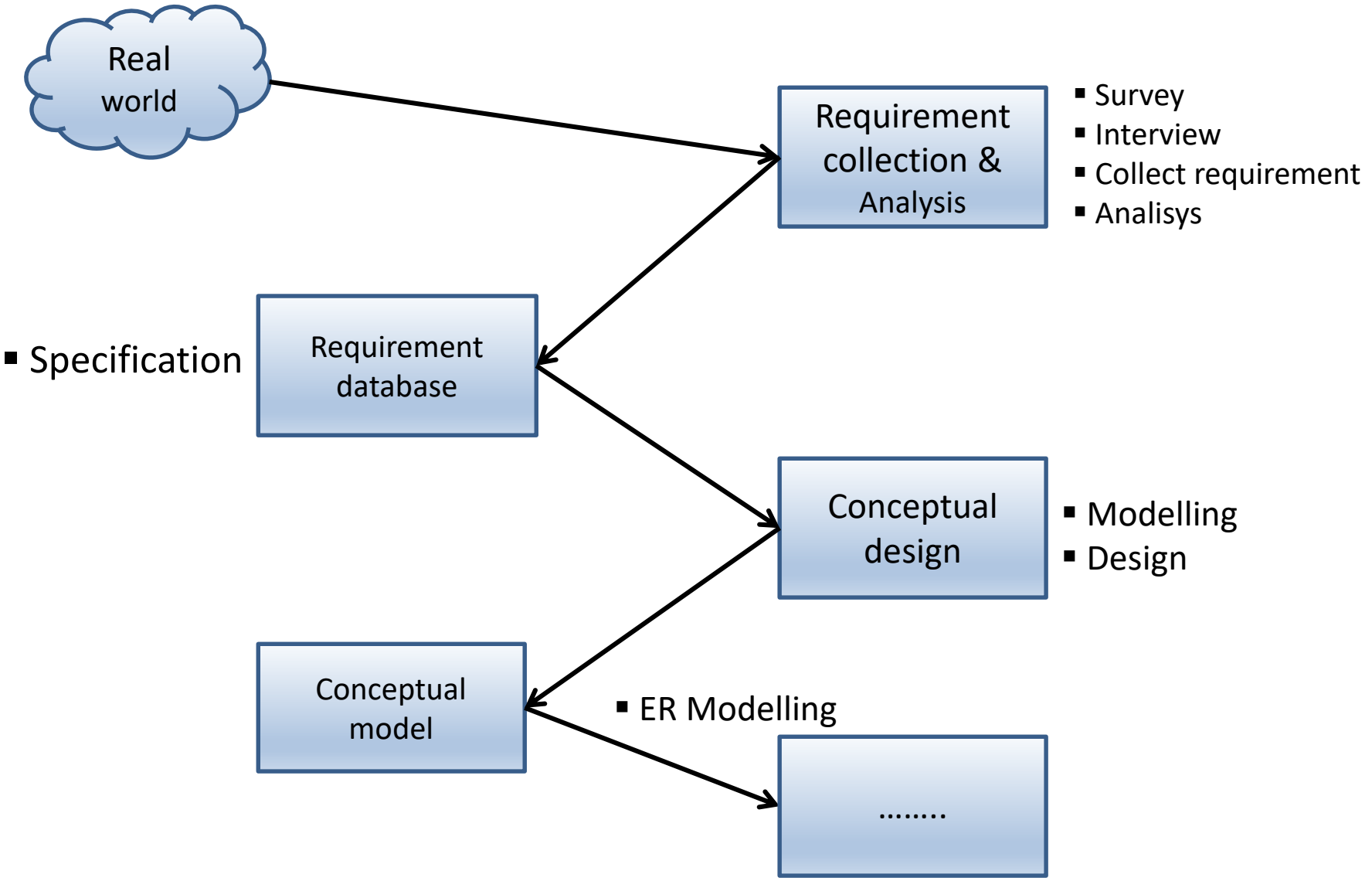
DBMS vs. RDBMS

DBMS	RDBMS
The concepts of relationships is missing in a DBMS. If it exits it is very less.	It is based on the concept Of relationships
Speed of operation is very slow	Speed of operation is very Fast
Hardware and Software requirements are minimum	Hardware and Software requirements are High
Platform used is normally DOS	Platform used can be any DOS, UNIX,VAX,VMS, etc
Uses concept of a file	Uses concept of table
DBMS normally use 3GL	RDBMS normally use a 4GL
Examples are dBase, FOXBASE, etc	Examples are ORACLE, INGRESS, SQL Server 2000 etc

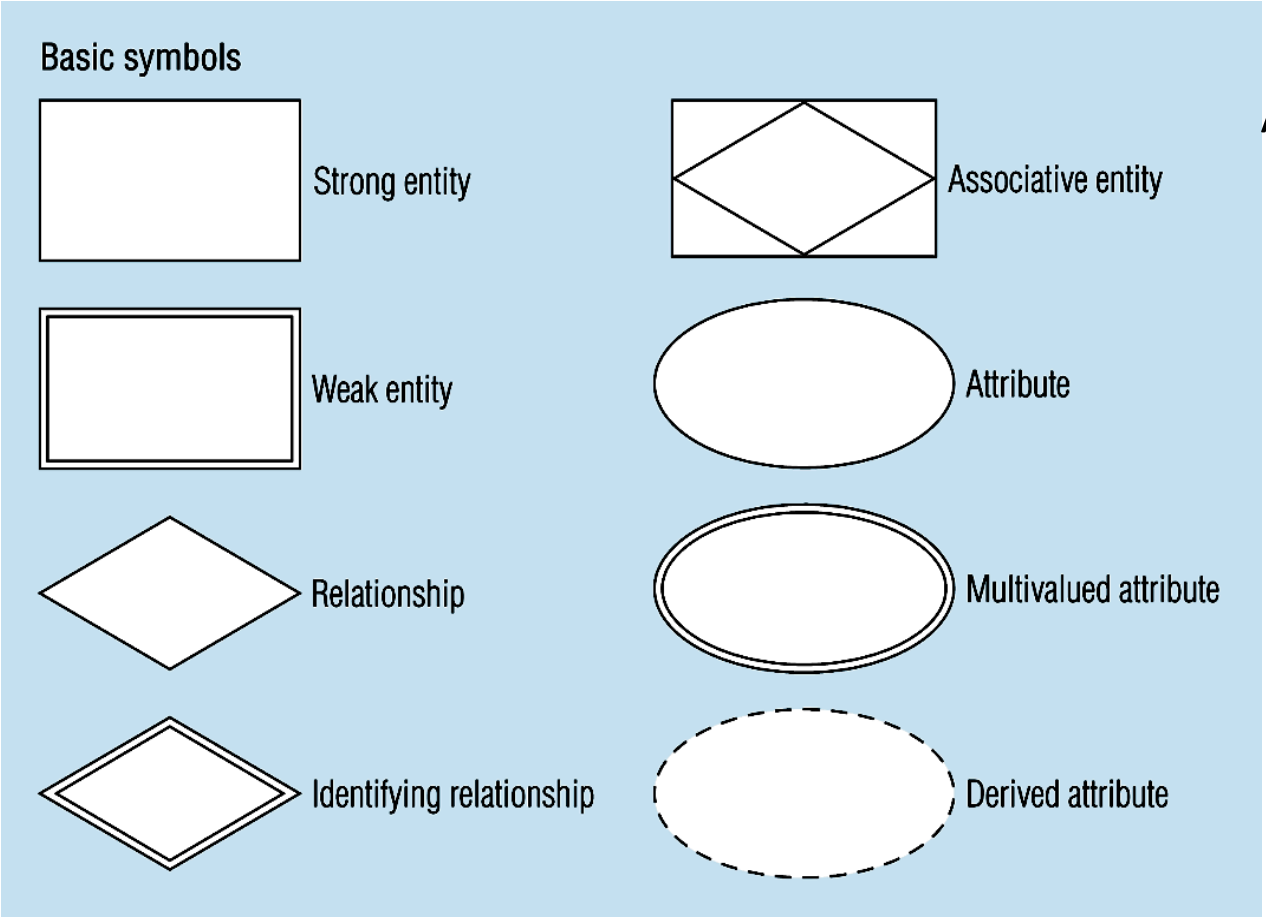
Section 3

ER MODEL

Design Process



Entity symbols

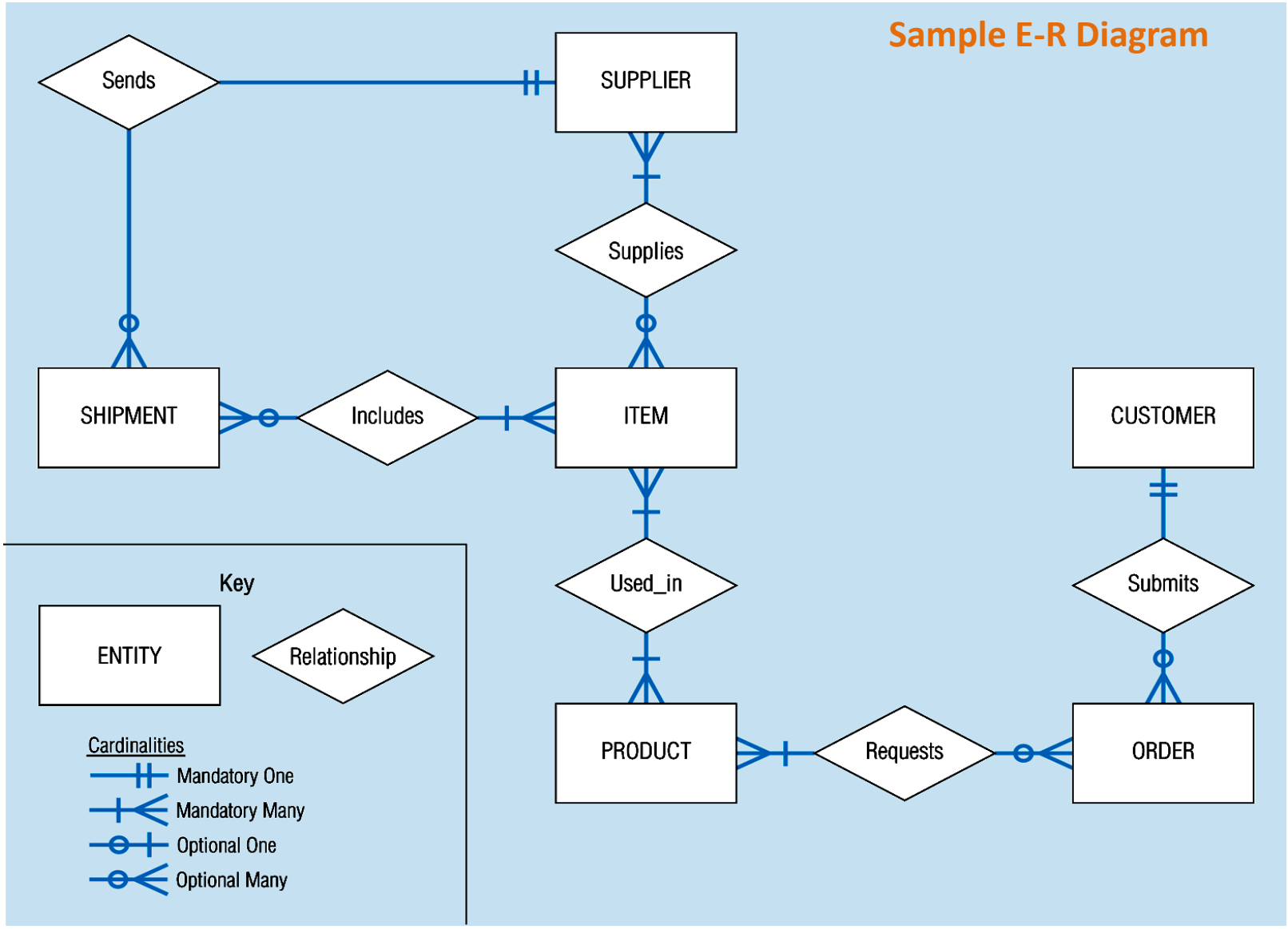


A special entity that is also a relationship

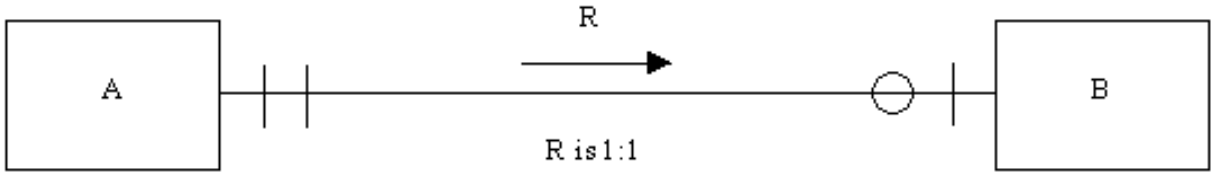
Relationship symbols

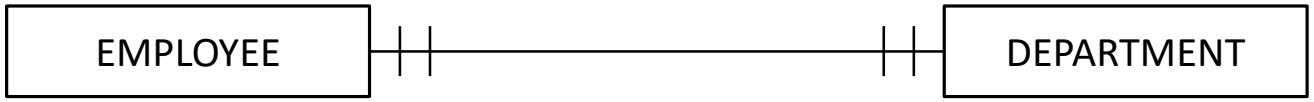
Attribute symbols

Sample E-R Diagram

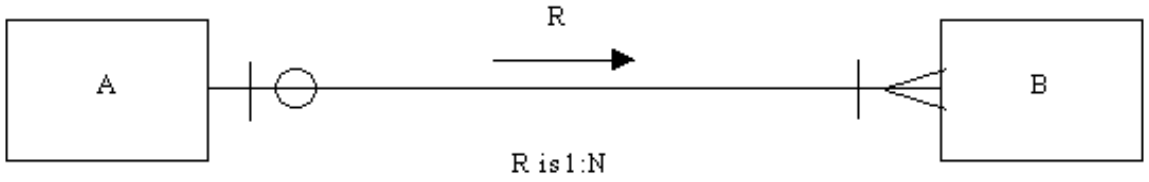


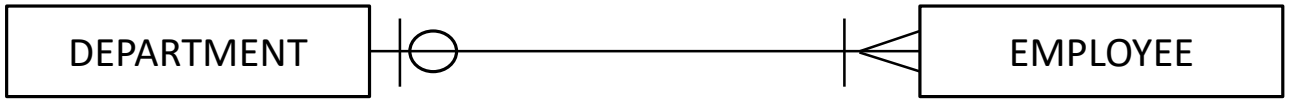
- **One – to – one:**



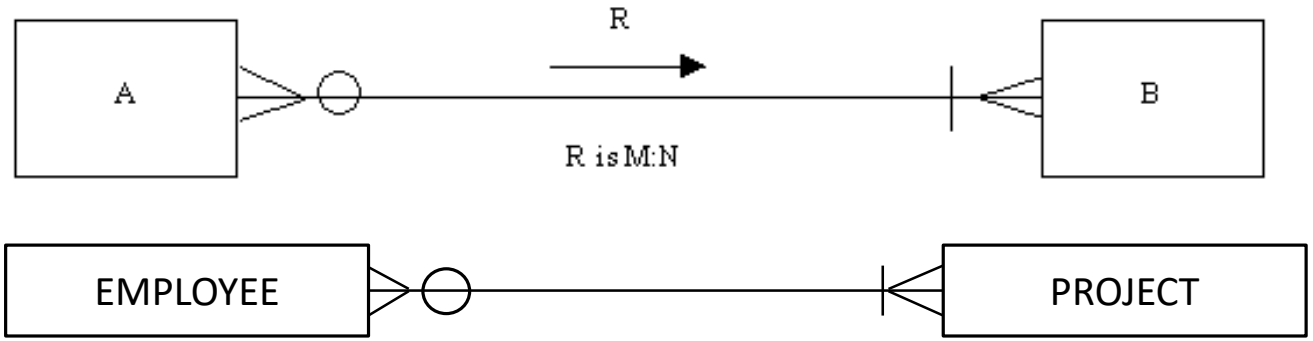


- **One – to – many:**

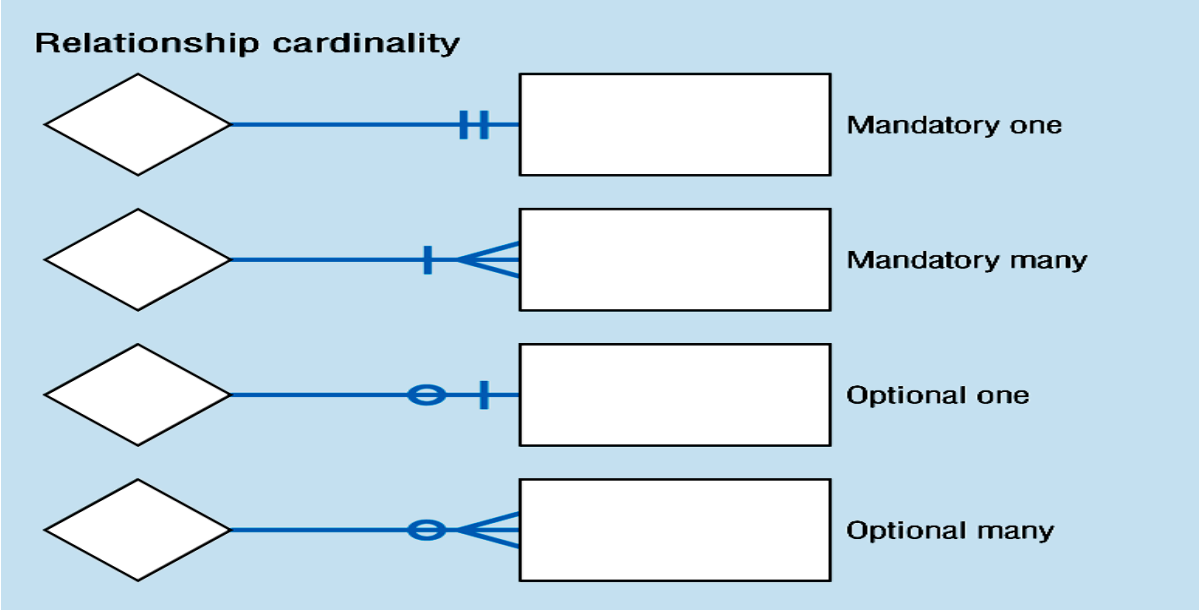


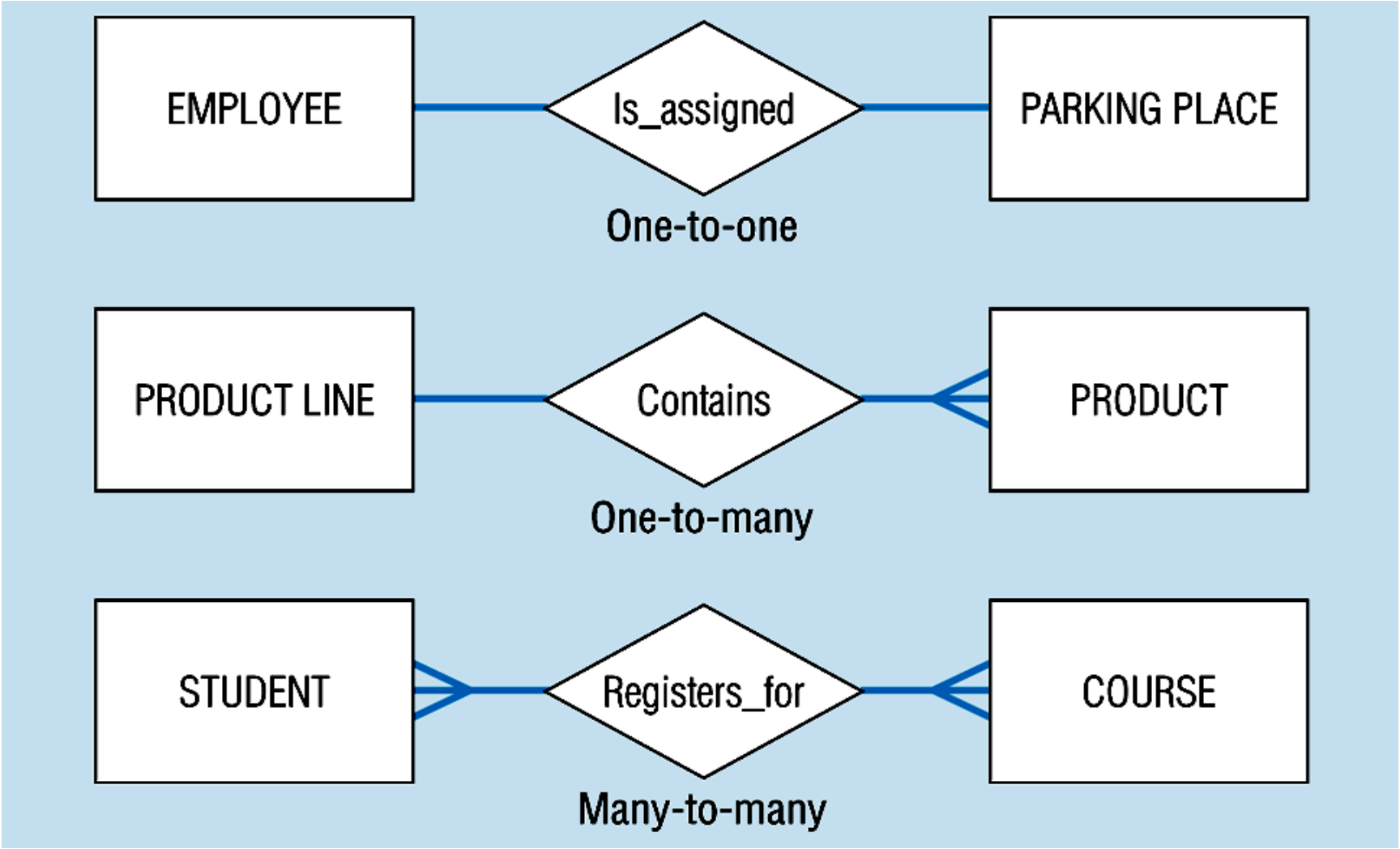


Many – to – many:



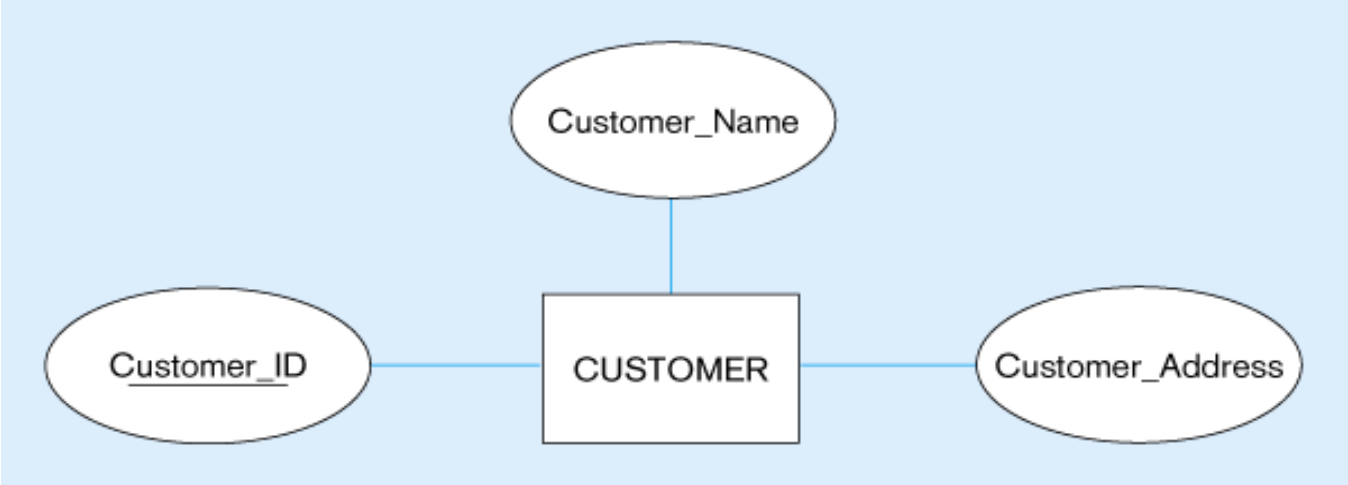
In which:



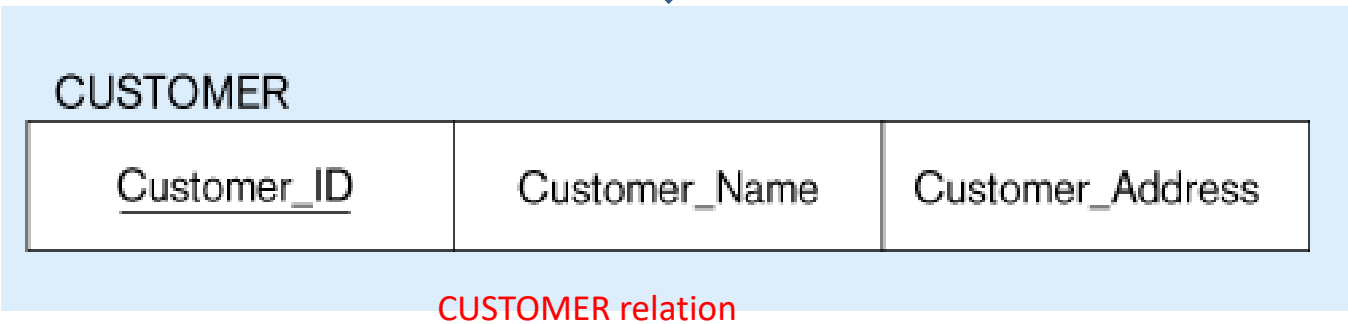


Converting ER Model to relational schema

Rule 1 - Convert entity type with simple attributes



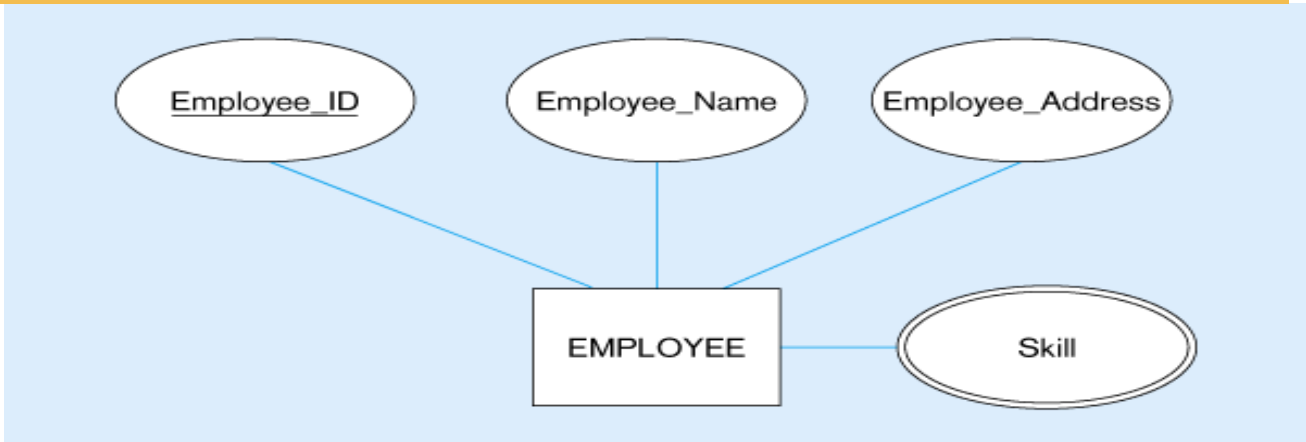
CUSTOMER entity type with simple attributes



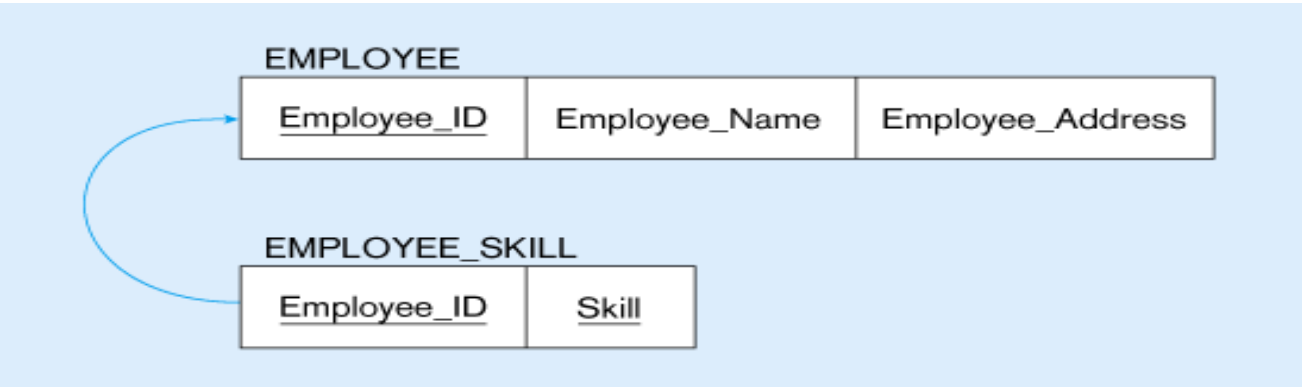
CUSTOMER relation

Converting ER Model to relational schema

Rule 2 - Convert Multivalue attribute



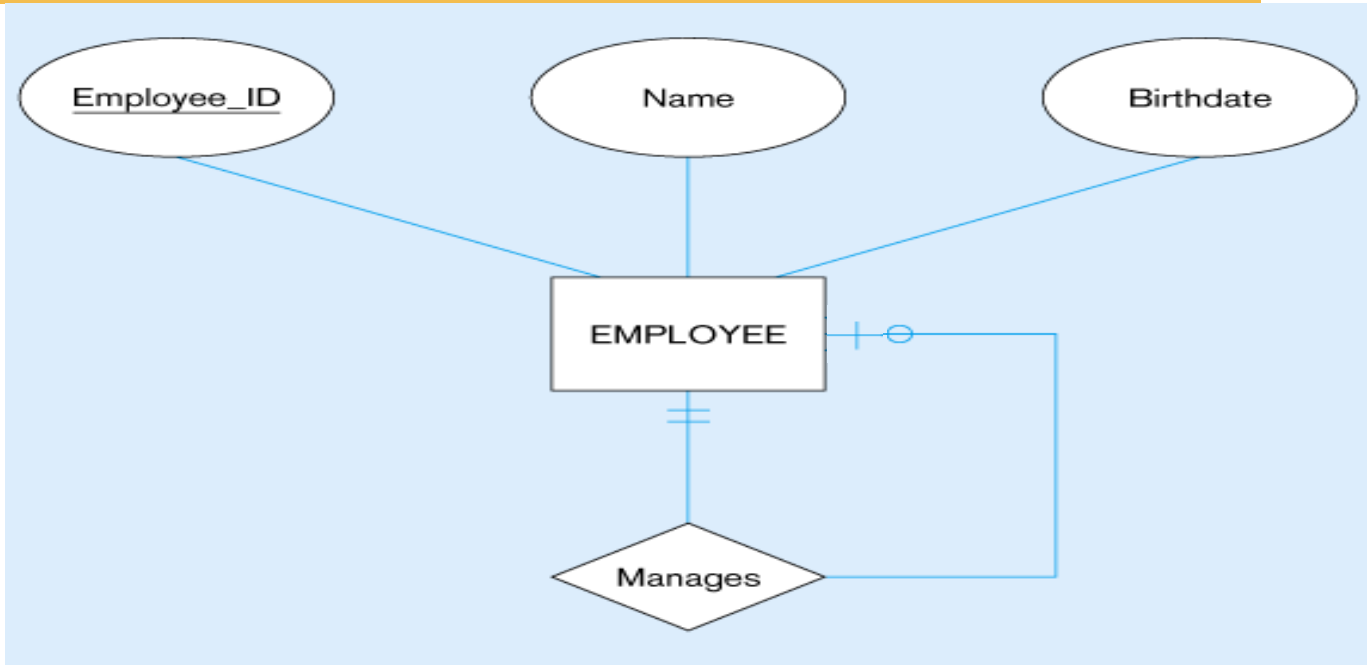
Multivalued attribute becomes a separate relation with foreign key



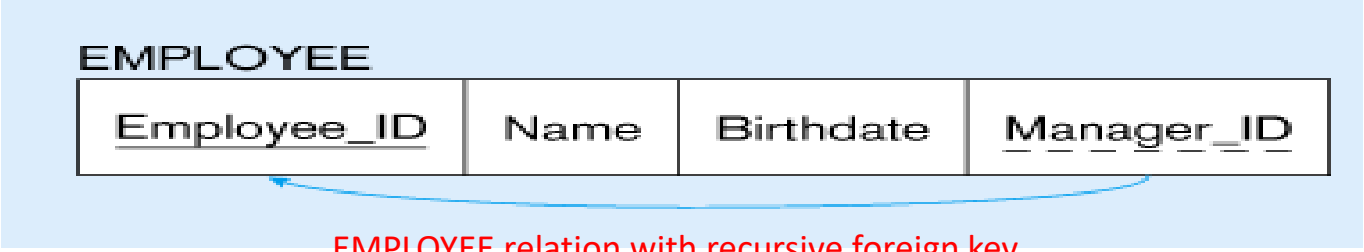
1–to–many relationship between original entity and new relation

Converting ER Model to relational schema

Rule 3 - Convert Unary relationship one to one



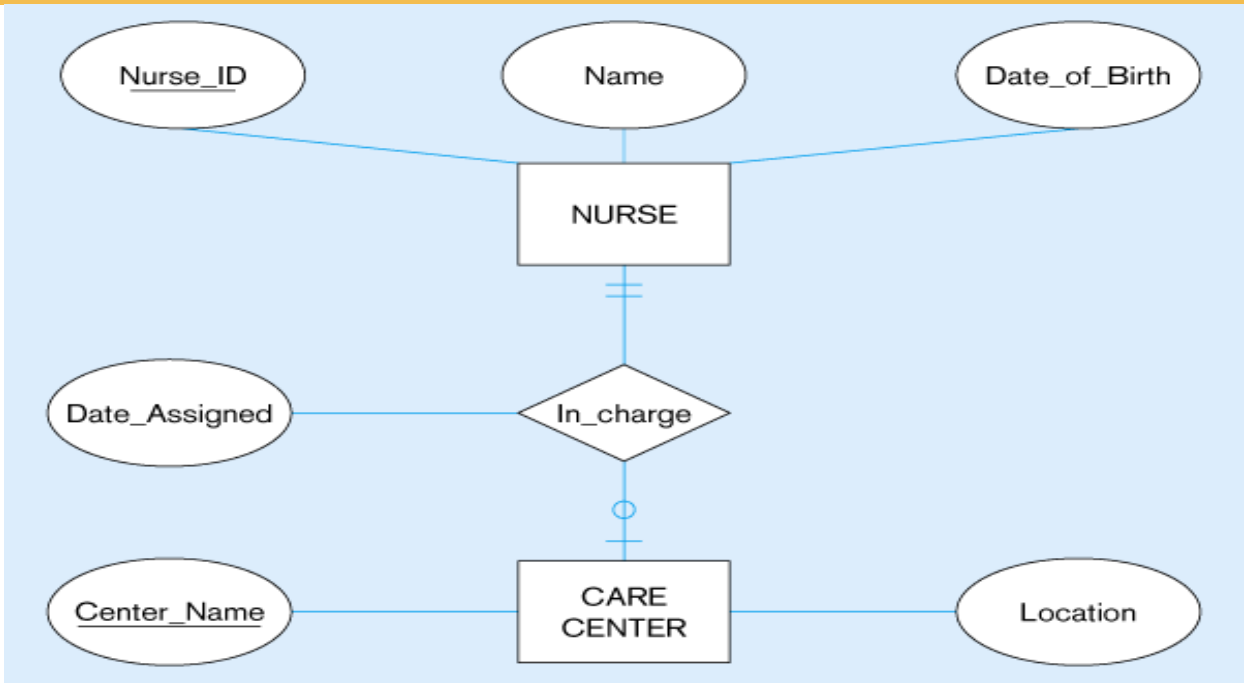
EMPLOYEE entity with Manages relationship



EMPLOYEE relation with recursive foreign key

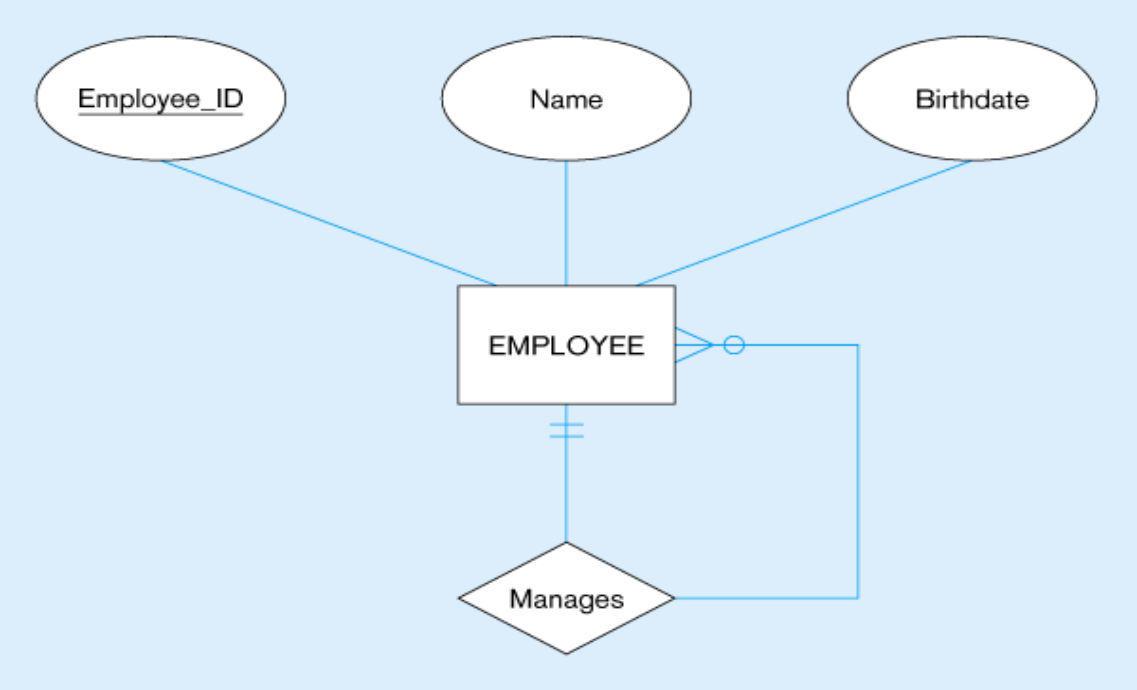
Converting ER Model to relational schema

Rule 4 – Convert binary relationship one to one

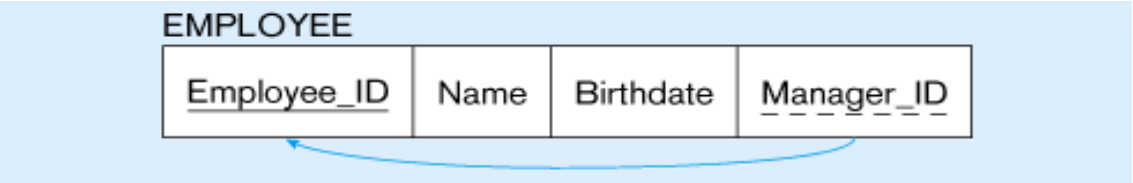


Converting ER Model to relational schema

Rule 5 – Convert Unary relationship one to many



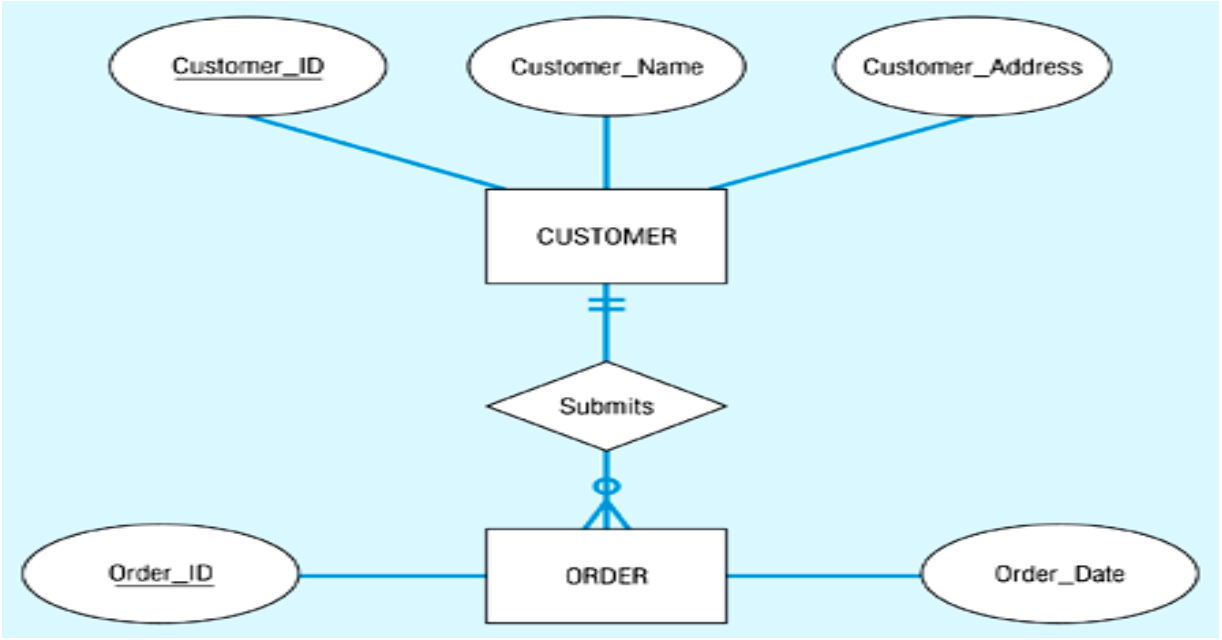
EMPLOYEE entity with Manages relationship



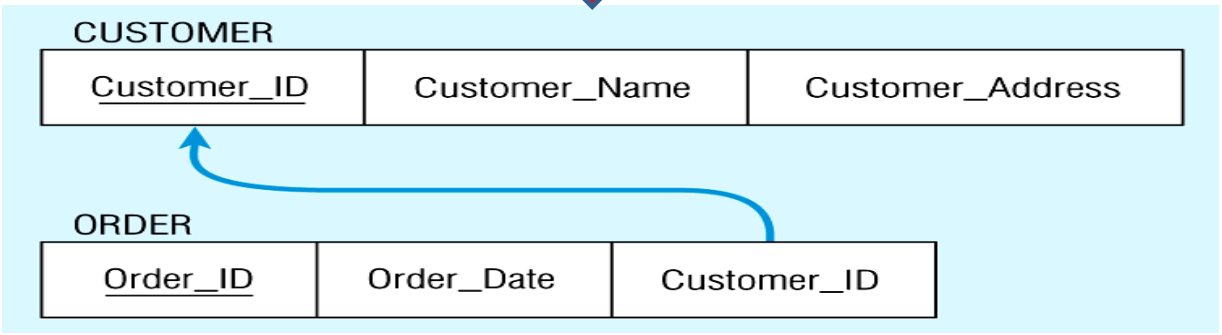
EMPLOYEE relation with recursive foreign key

Converting ER Model to relational schema

Rule 6 – Convert Binary relationship one to many



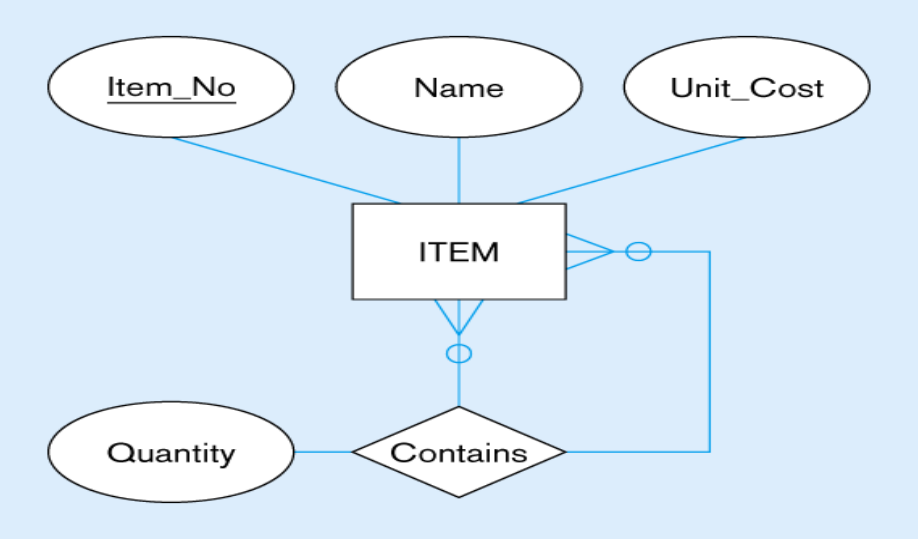
Note the mandatory one



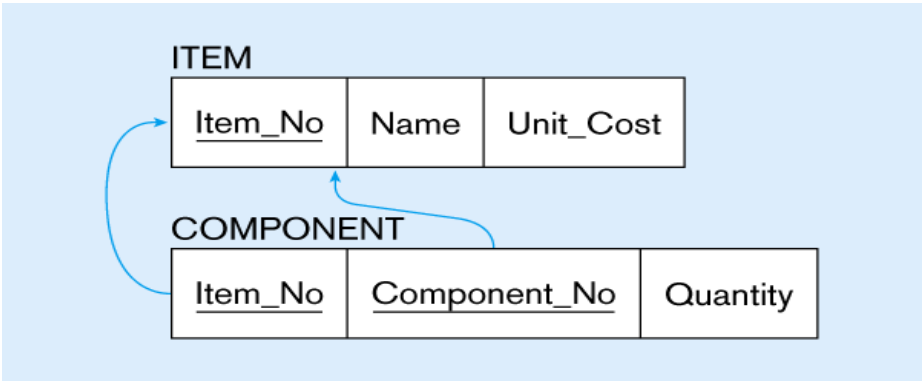
Again, no null value in the foreign key…this is because of the mandatory minimum cardinality

Converting ER Model to relational schema

Rule 7 – Convert Unary relationship many to many



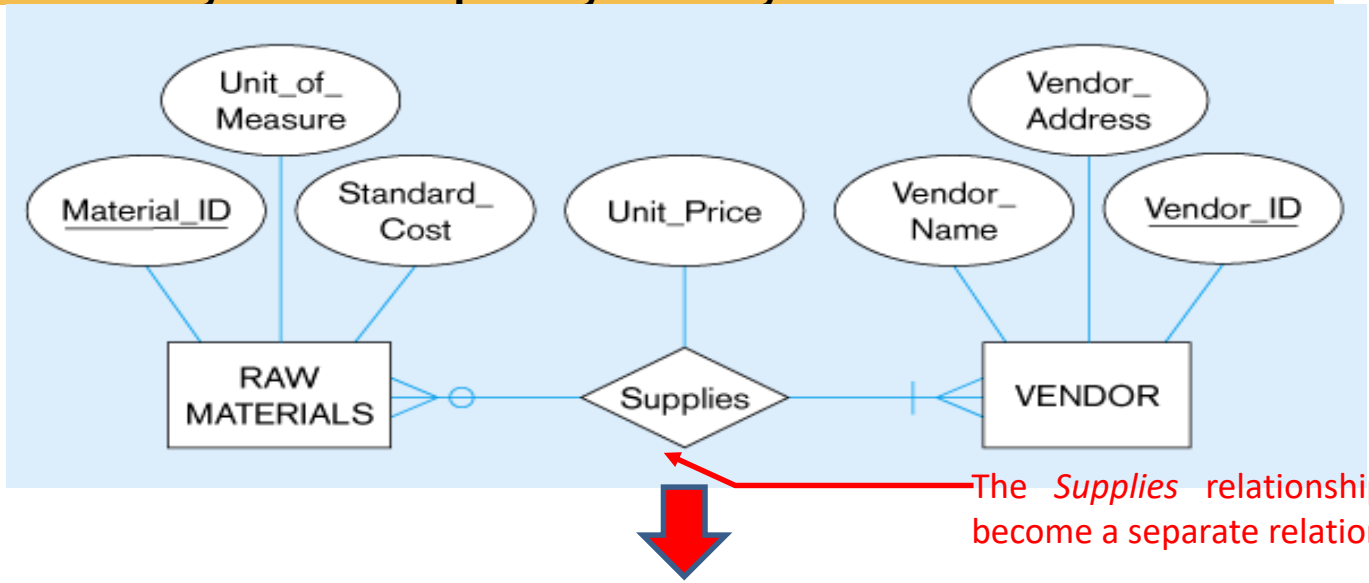
Bill-of-materials relationships (M:N)

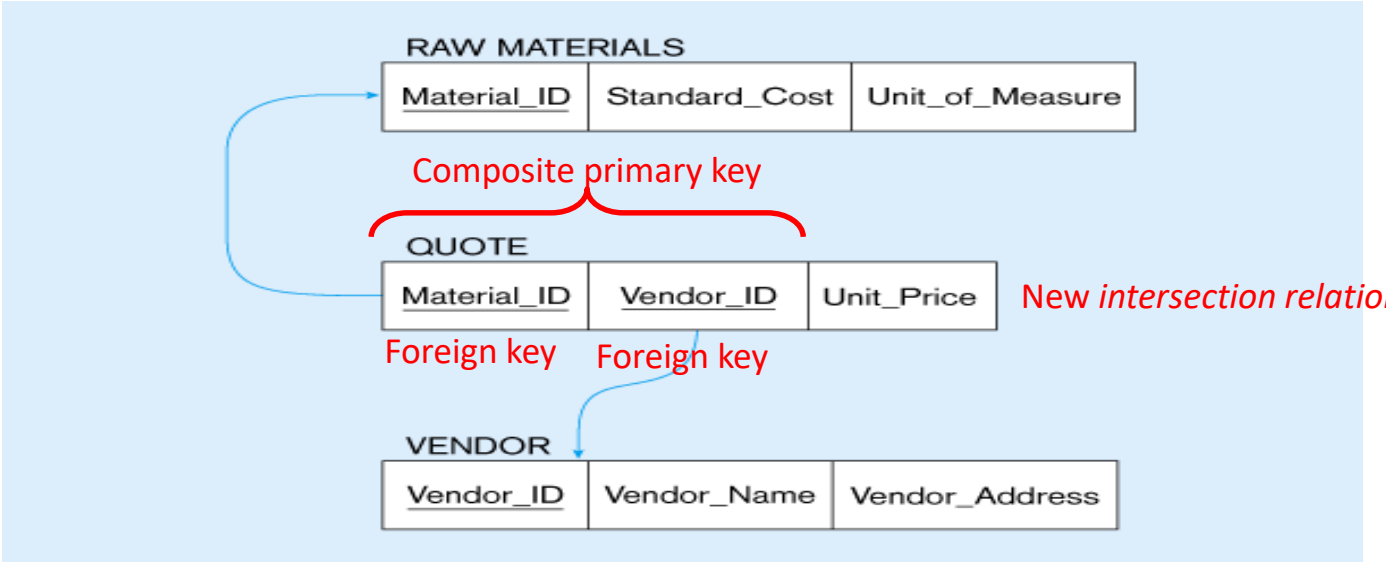


ITEM and COMPONENT relations

Converting ER Model to relational schema

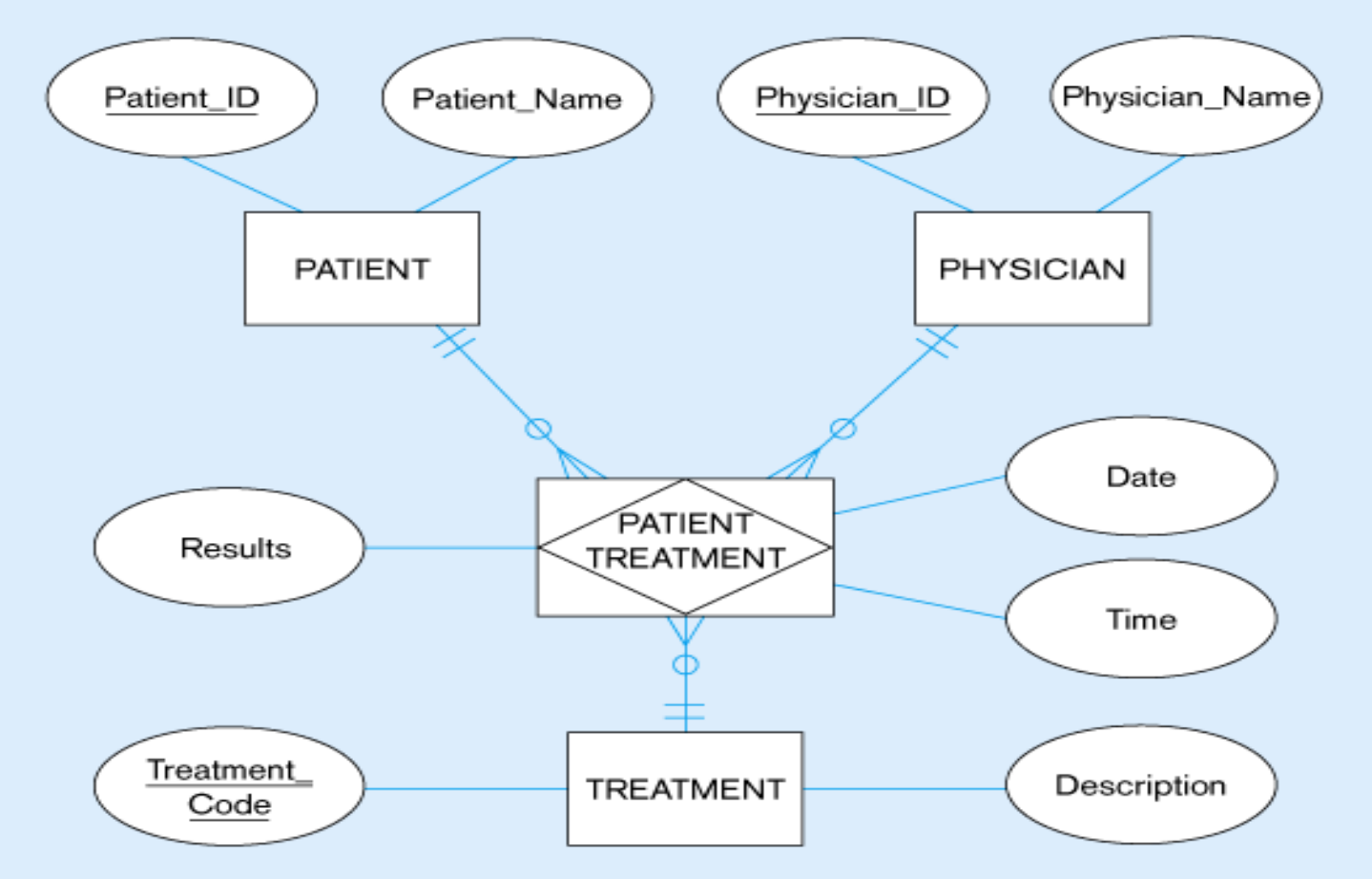
Rule 8 – Convert Binary relationship many to many





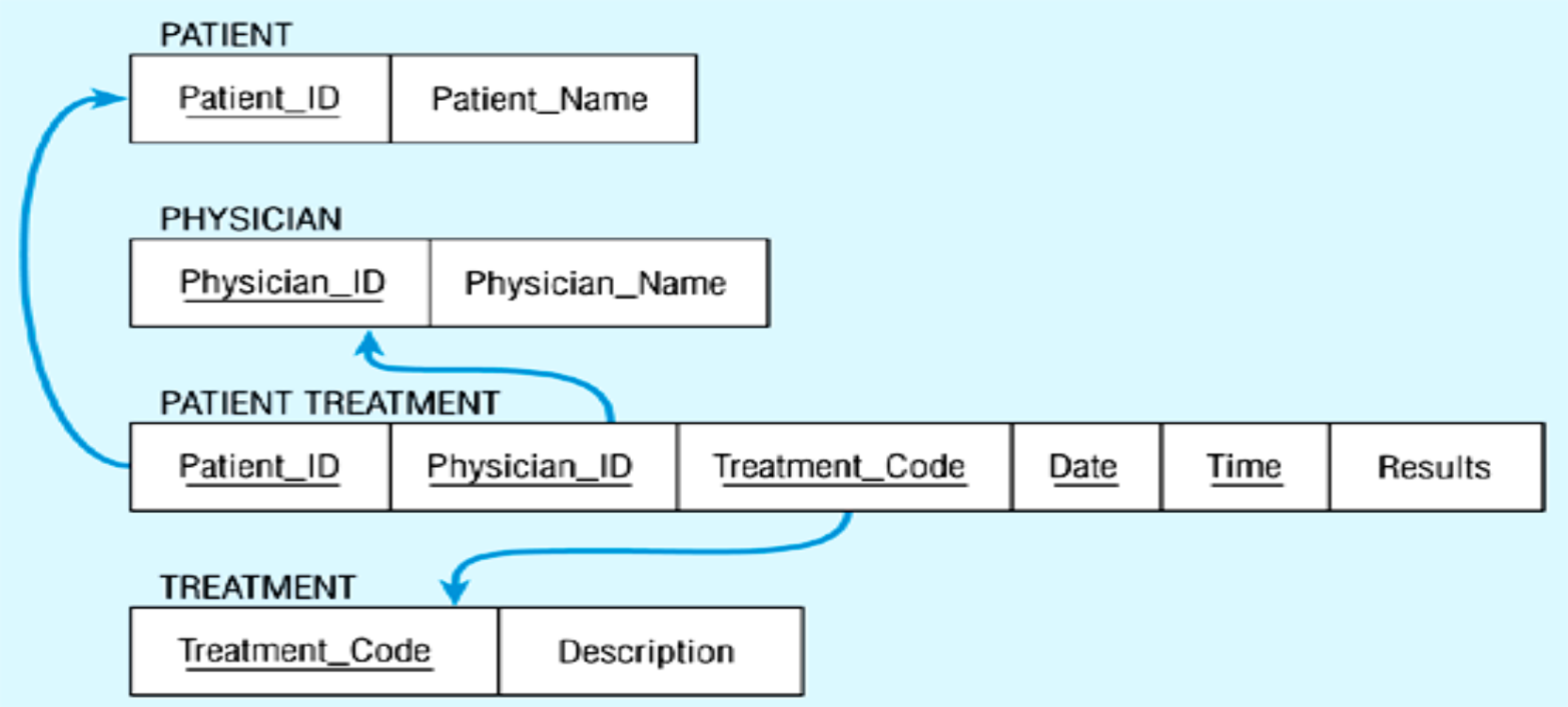
Converting ER Model to relational schema

Another - Convert Ternary relationship



Converting ER Model to relational schema

Another- Convert Ternary relationship (2)



Summary

- ✓ SQL Overview
  - SQL, SQL Process, SQL Command
- ✓ The Relational Database
  - Table, Field, Record, Schema
- ✓ RDBMS Concepts
  - RDBMS, RDBMS vs DBMS
- ✓ ER Model
  - Design Process, **Notation,** Converting ER Model to relational schema