EIE 3112
Relational Models and ER Diagrams

Dr Pauli Lai

T. Connolly and C. Begg, "*Database Systems: A Practical Approach to Design, Implementation, and Management*," 6th Edition, Chapter 4, Pearson, 2015. (5th Edition is also fine)

Intended Learning Outcomes (ILO)

Subject ILO to achieve:

A: Professional/academic knowledge and skills

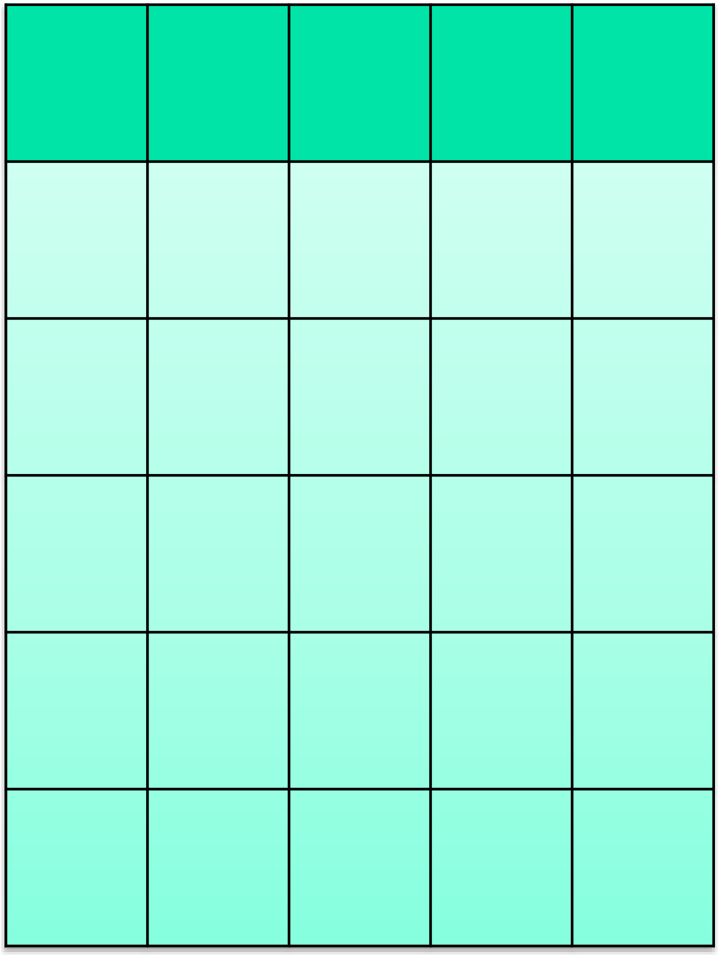
1. Database design, development, and programming

Topics:

- Terminology of relational model.
- How tables are used to represent data.
- How to identify CK, PK, and FKs.
- Meaning of entity integrity and referential integrity
- ER Diagrams

Relational Model Terminology

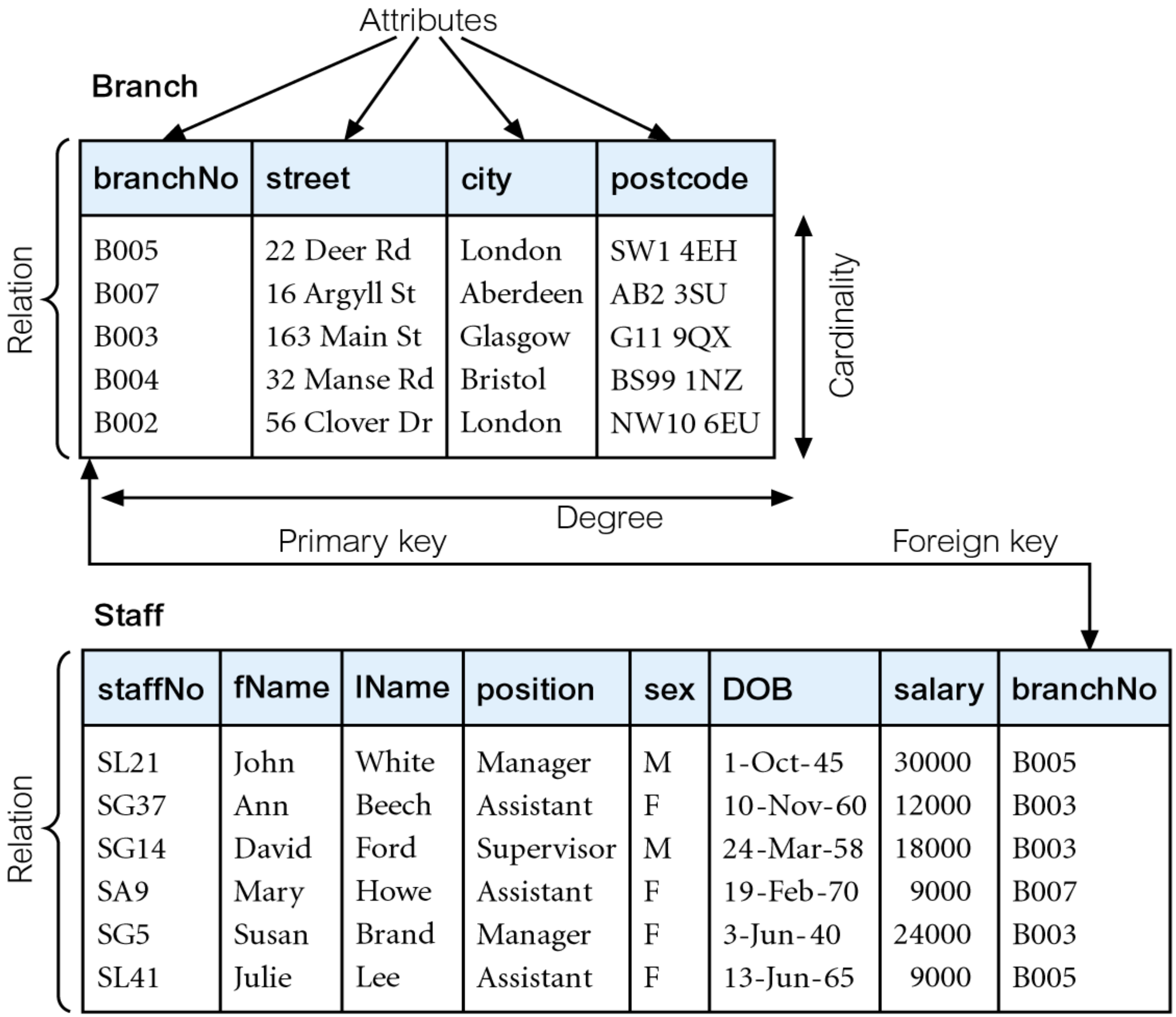
- A relation is a table with columns and rows.
  - Only applies to logical structure of the database, not the physical structure.
- Attribute is a named column of a relation.
- Domain is the set of allowable values for one or more attributes.



Relational Model Terminology

- Tuple is a row of a relation.
- Degree is the number of attributes in a relation.
- Cardinality is the number of tuples in a relation.
- Relational Database is a collection of relations with distinct relation names.

Branch and Staff Relations



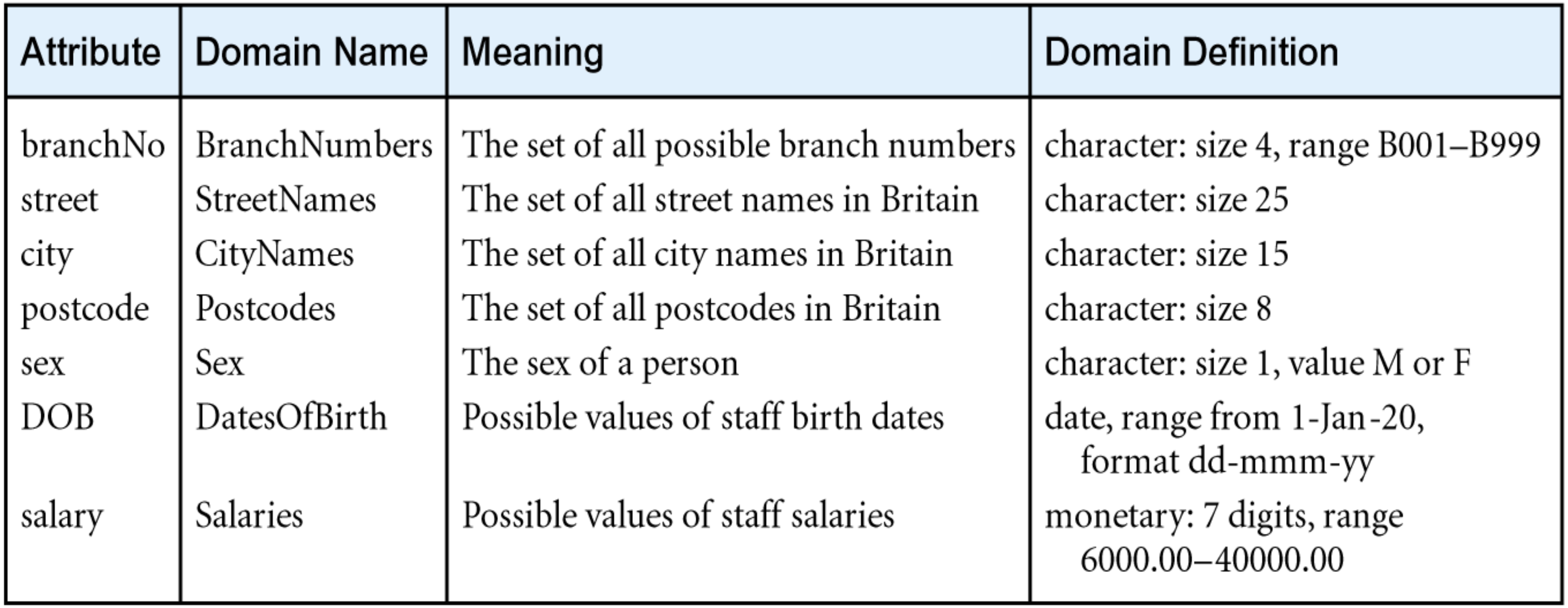
Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

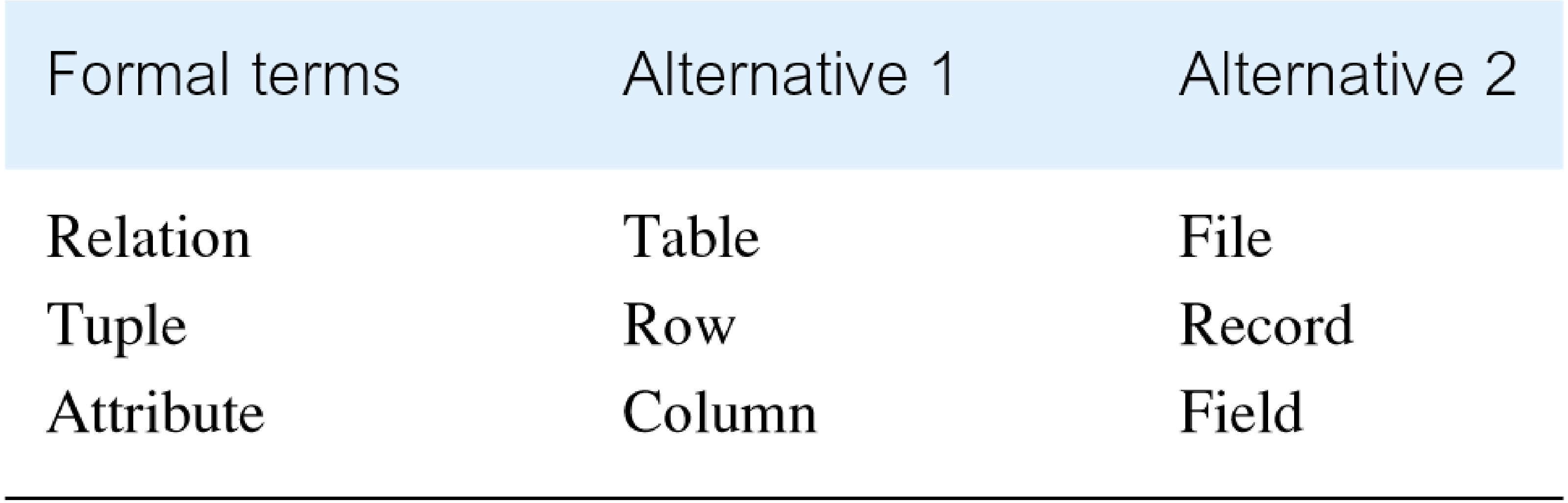
staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Example of Attribute Domains



Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8
sex	Sex	The sex of a person	character: size 1, value M or F
DOB	DatesOfBirth	Possible values of staff birth dates	date, range from 1-Jan-20, format dd-mmm-yy
salary	Salaries	Possible values of staff salaries	monetary: 7 digits, range 6000.00–40000.00

Alternative Terminology



Formal terms	Alternative 1	Alternative 2
Relation	Table	File
Tuple	Row	Record
Attribute	Column	Field

Class Activity: Q&A

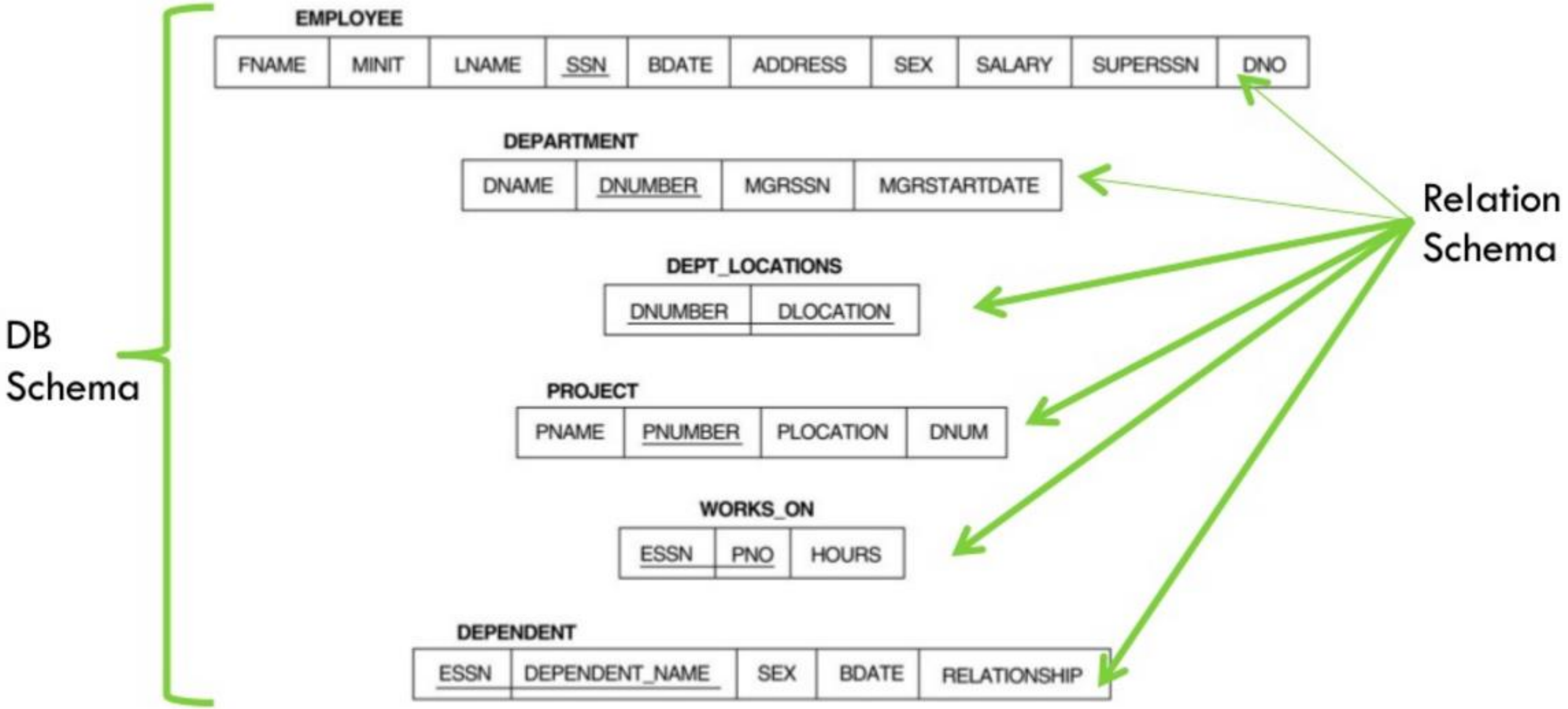
- When I call your name, please turn on your mic to answer the question.
- Example questions related to the relational model terminology:
  - How many tuples are there in relation Branch?
  - What is the Cardinality of table Staff?
  - How many attributes are there in file Staff?
  - What is the Degree of relation Branch?
  - How many fields are there in table Staff?
  - What is the domain for staffNo in relation Staff?
  - What is the domain for position in relation Staff?
  - Can you suggest one more column for Staff?
  - Can you suggest one more row for Branch?

Database Relations

- Relation schema
  - Named relation defined by a set of attribute and domain name pairs.
- Relational database schema
  - Set of relation schemas, each with a distinct name.

Relational Database schema

- A sets of relation schema.
- The example shows the database schema for COMPANY = {EMPLOYEE, DEPARTMENT, DEPENDENT, PROJECT}



DB Schema

EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

DEPT_LOCATIONS

DNUMBER	DLOCATION

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

WORKS_ON

ESSN	PNO	HOURS

DEPENDENT

ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP

Relation Schema

Properties of Relations

- Relation name is distinct from all other relation names in relational DB schema.
- Each cell of relation contains exactly one atomic (single) value.
- Each attribute has a distinct name.
- Values of an attribute are all from the same domain.

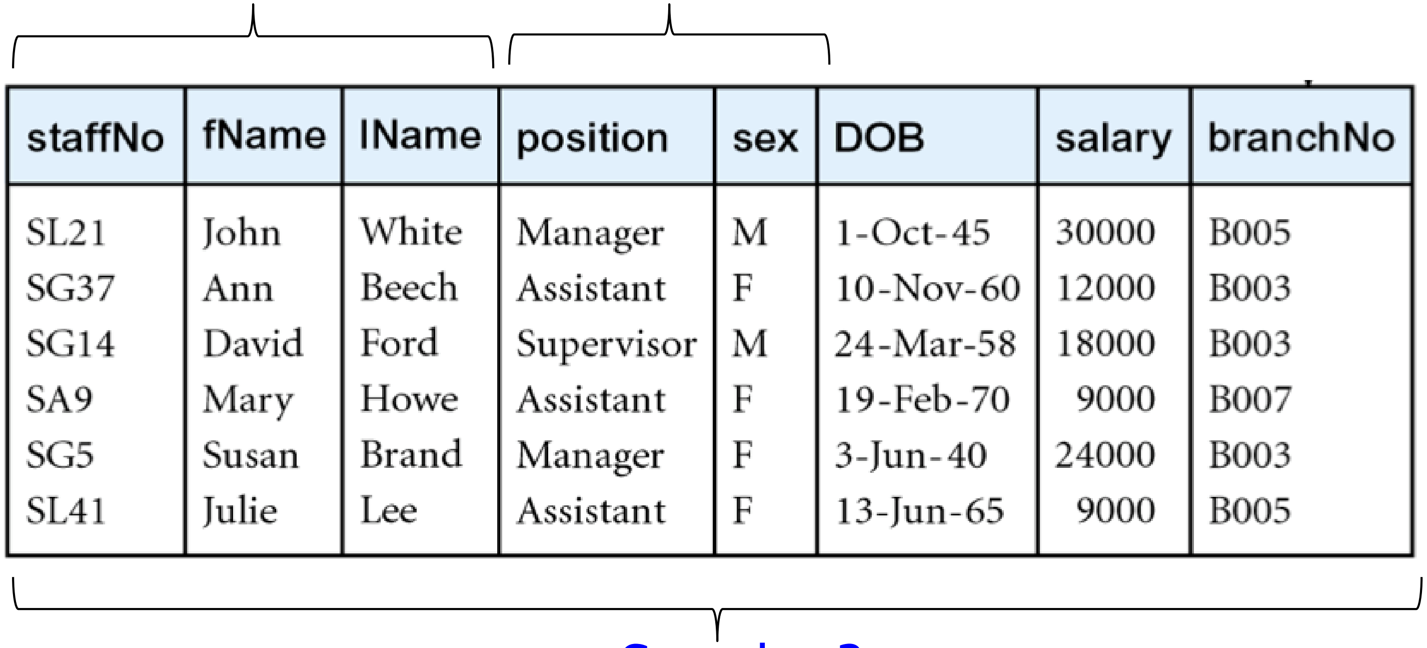
Properties of Relations

- Each tuple is distinct; there are no duplicate tuples.
- Order of attributes has no significance.
- Order of tuples has no significance, theoretically.

Relational Keys

- **Superkey**
  - An attribute, or set of attributes, that uniquely identifies a tuple within a relation.
  - A superkey may contain additional attributes that are not necessary for unique identification.

Superkey? Superkey?



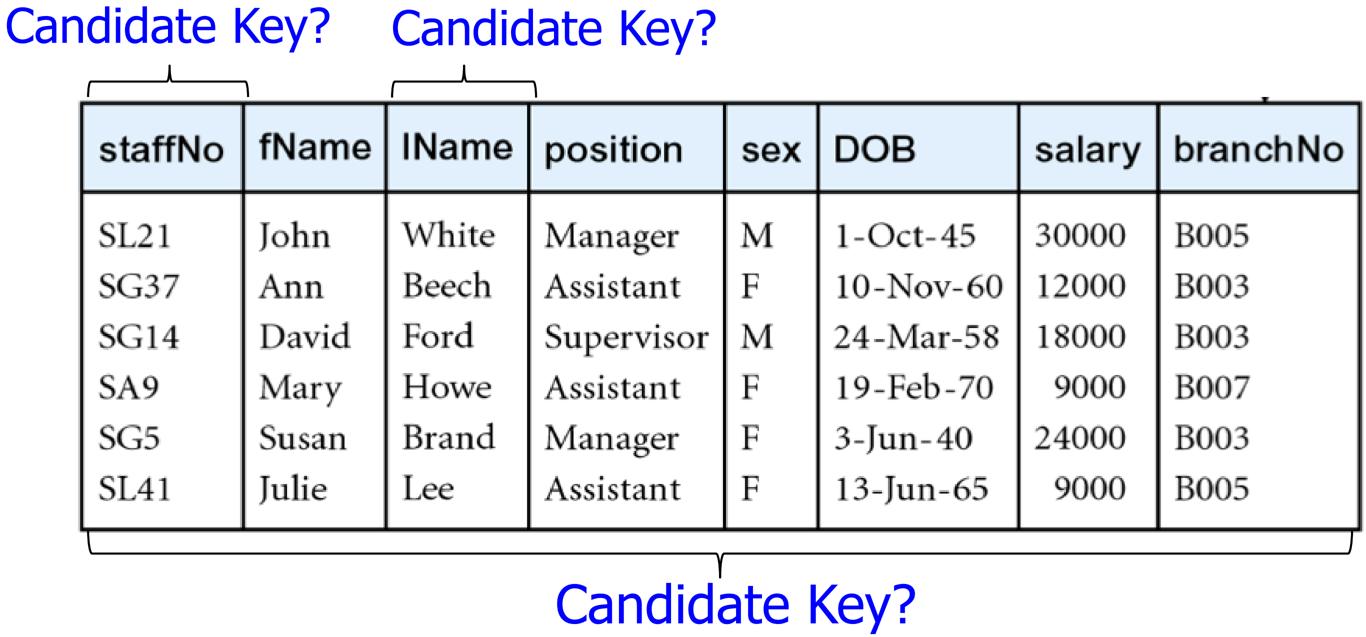
staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Superkey?

Relational Keys

Candidate Key

- Superkey (*K*) such that no proper subset is a superkey within the relation (minimum superkey)
- In each tuple of relation *R*, values of candidate *K* uniquely identify that tuple (uniqueness).
- No proper subset of *K* has the uniqueness property (irreducibility).
- Identifying a candidate key requires that we know the “real-world” meaning of the attributes

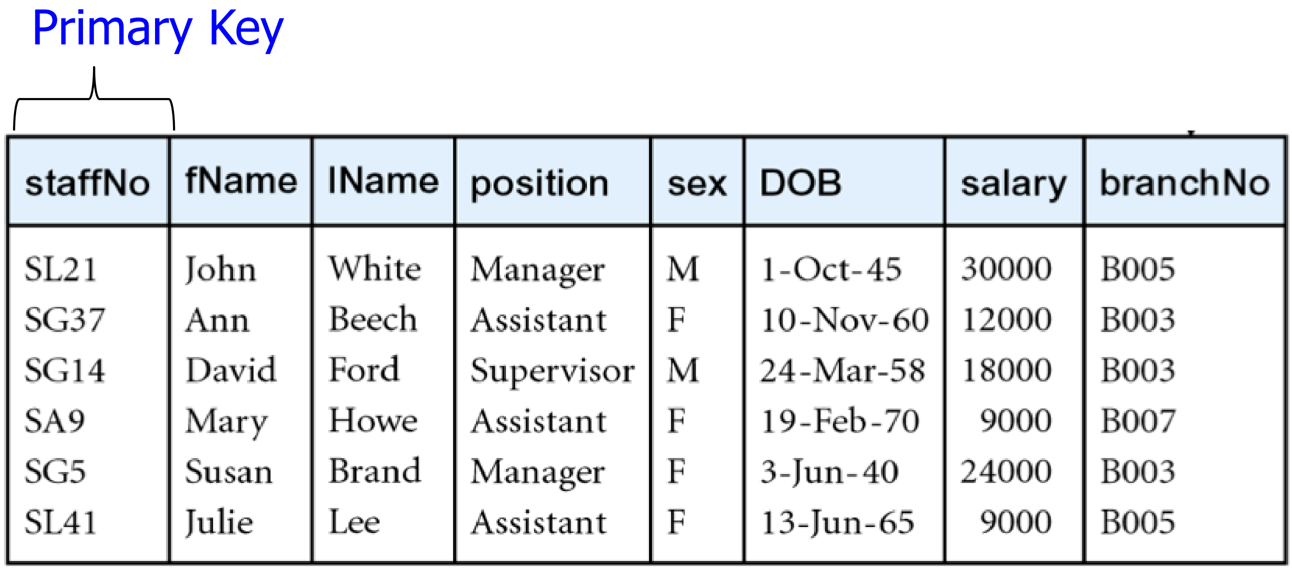

Candidate Key?
Candidate Key?
Candidate Key?

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Relational Keys

- **Primary Key**
  - Candidate key selected to identify tuples uniquely within a relation.

- **Alternate Keys**
  - Candidate keys that are not selected to be primary key.

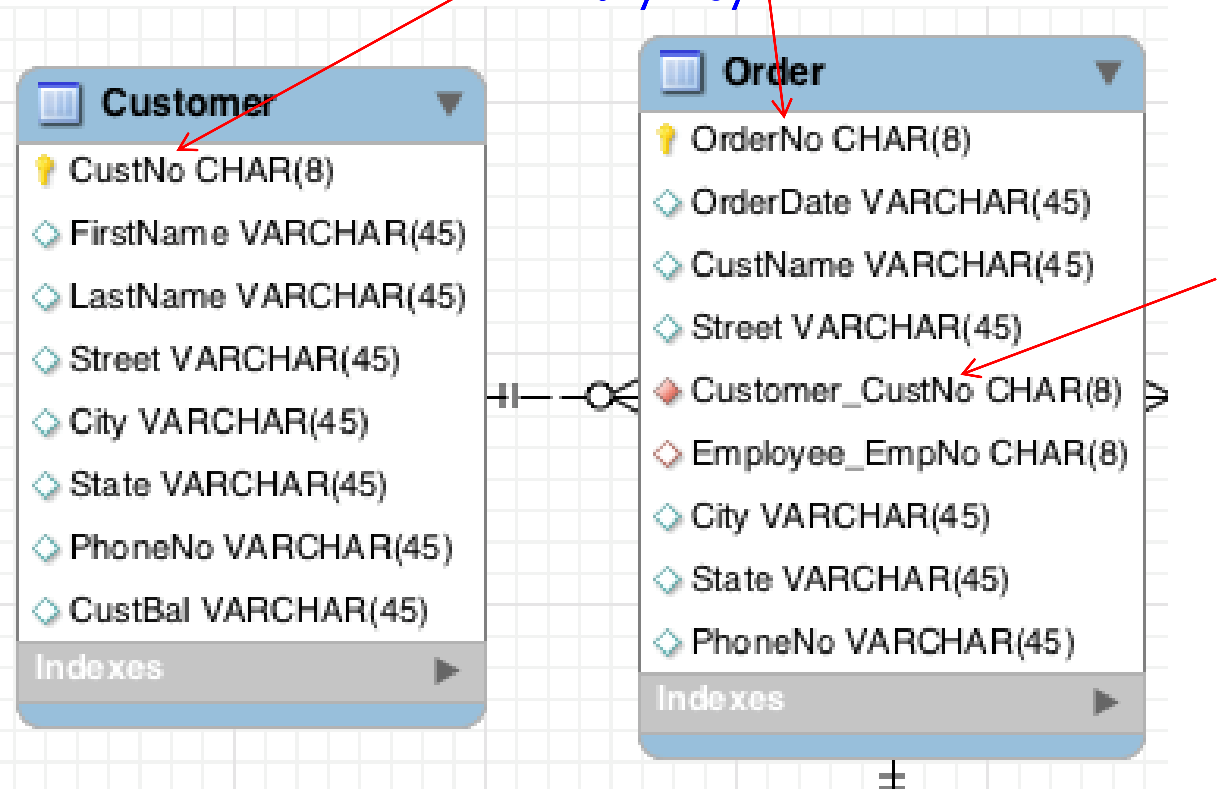

Primary Key

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Relational Keys

- **Foreign Key**
  - Attribute, or set of attributes, within one relation that matches candidate key of some (possibly same) relation.

Primary Key



Customer

- CustNo CHAR(8)
- FirstName VARCHAR(45)
- LastName VARCHAR(45)
- Street VARCHAR(45)
- City VARCHAR(45)
- State VARCHAR(45)
- PhoneNo VARCHAR(45)
- CustBal VARCHAR(45)

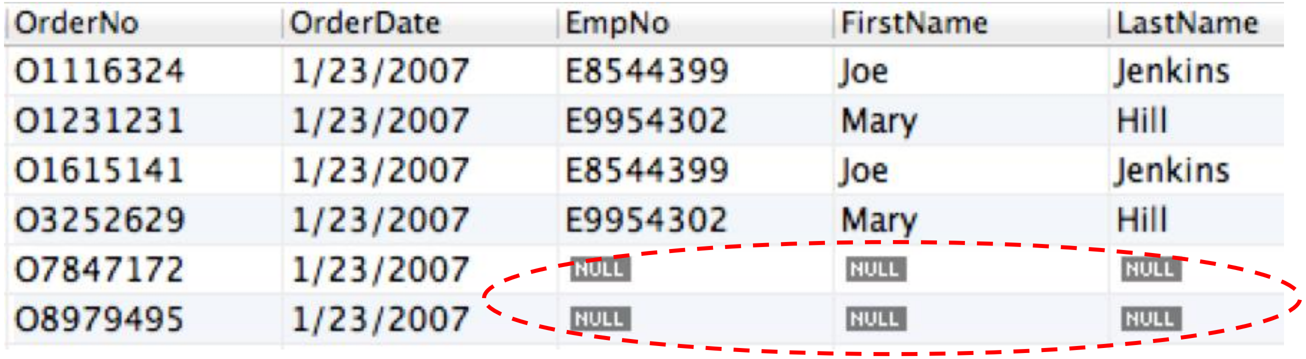
Order

- OrderNo CHAR(8)
- OrderDate VARCHAR(45)
- CustName VARCHAR(45)
- Street VARCHAR(45)
- Customer_CustNo CHAR(8)
- Employee_EmpNo CHAR(8)
- City VARCHAR(45)
- State VARCHAR(45)
- PhoneNo VARCHAR(45)

Foreign Key

Integrity Constraints

- **Null**
  - Represents value for an attribute that is currently unknown or not applicable for tuple.
  - Deals with incomplete or exceptional data.
  - Represents the absence of a value and is not the same as zero or spaces, which are values.



OrderNo	OrderDate	EmpNo	FirstName	LastName
O1116324	1/23/2007	E8544399	Joe	Jenkins
O1231231	1/23/2007	E9954302	Mary	Hill
O1615141	1/23/2007	E8544399	Joe	Jenkins
O3252629	1/23/2007	E9954302	Mary	Hill
O7847172	1/23/2007	NULL	NULL	NULL
O8979495	1/23/2007	NULL	NULL	NULL

Integrity Constraints

- **Entity Integrity**
  - Each relation has column(s) with unique values
  - No two rows have the same value
  - No attribute of a primary key can be null
  - Can be achieved by having a primary key

- **Referential Integrity**
  - Column values in one table must match column values in a related table.

Views

- **Base Relation**
  - Named relation corresponding to an entity in conceptual schema, whose tuples are physically stored in database.

- **View**
  - Dynamic result of one or more relational operations operating on base relations to produce another relation.

Views

Allows each user to have his or her own view of the database.

A view is essentially some subset of the database.

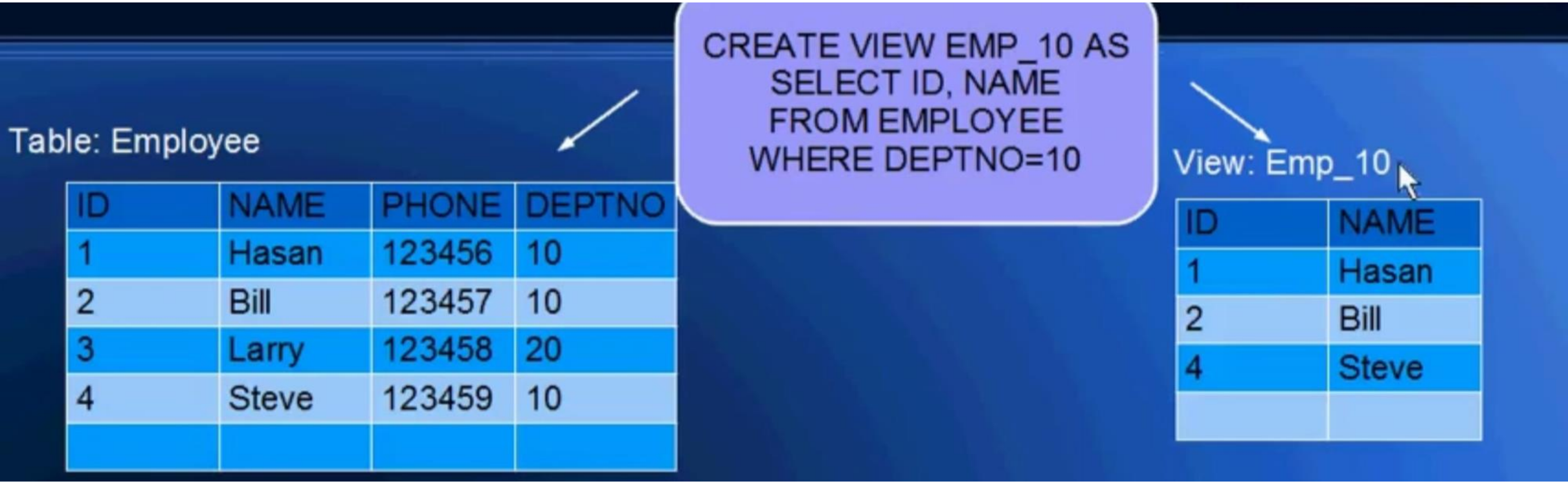


Table: Employee

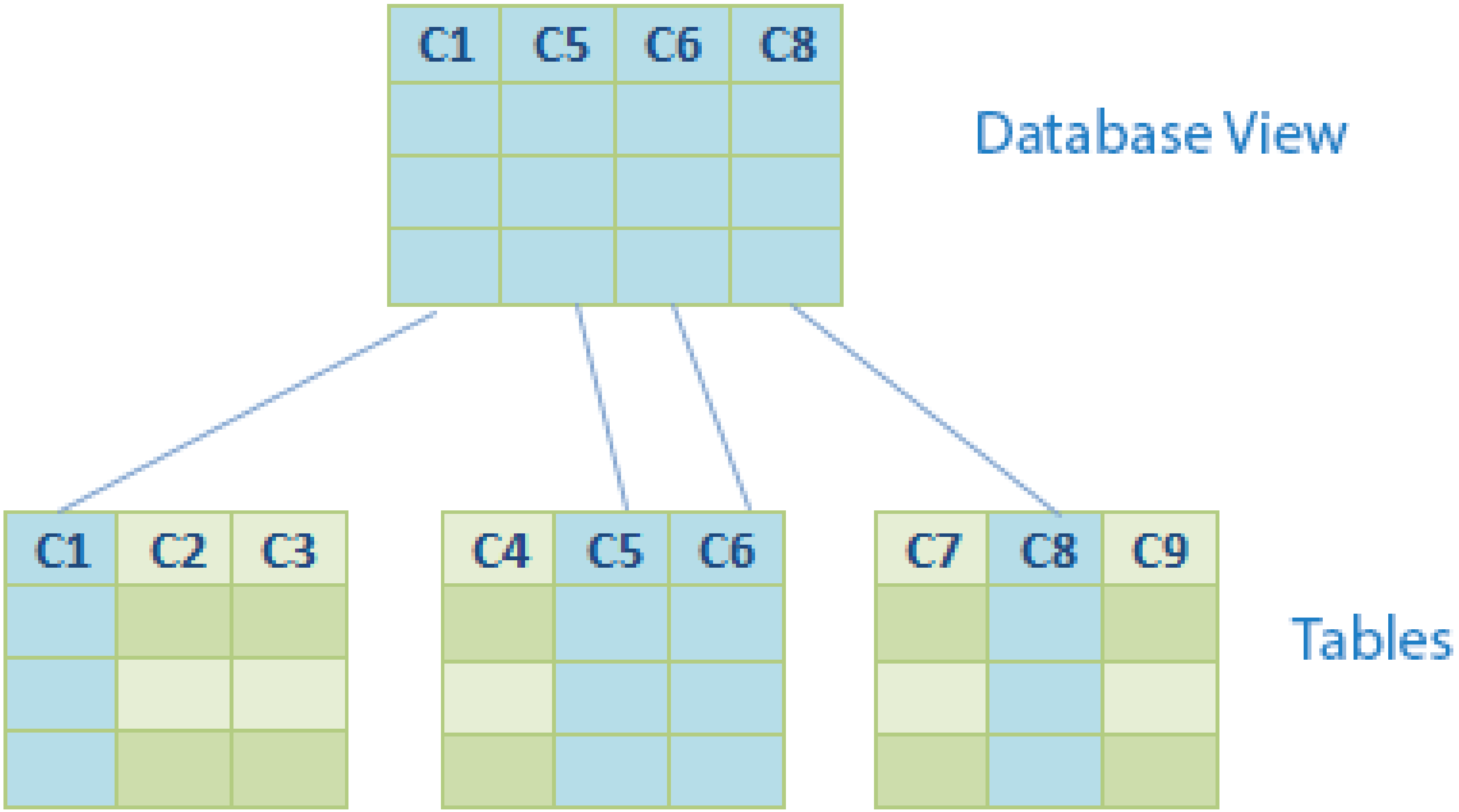
ID	NAME	PHONE	DEPTNO
1	Hasan	123456	10
2	Bill	123457	10
3	Larry	123458	20
4	Steve	123459	10

```
CREATE VIEW EMP_10 AS
SELECT ID, NAME
FROM EMPLOYEE
WHERE DEPTNO=10
```

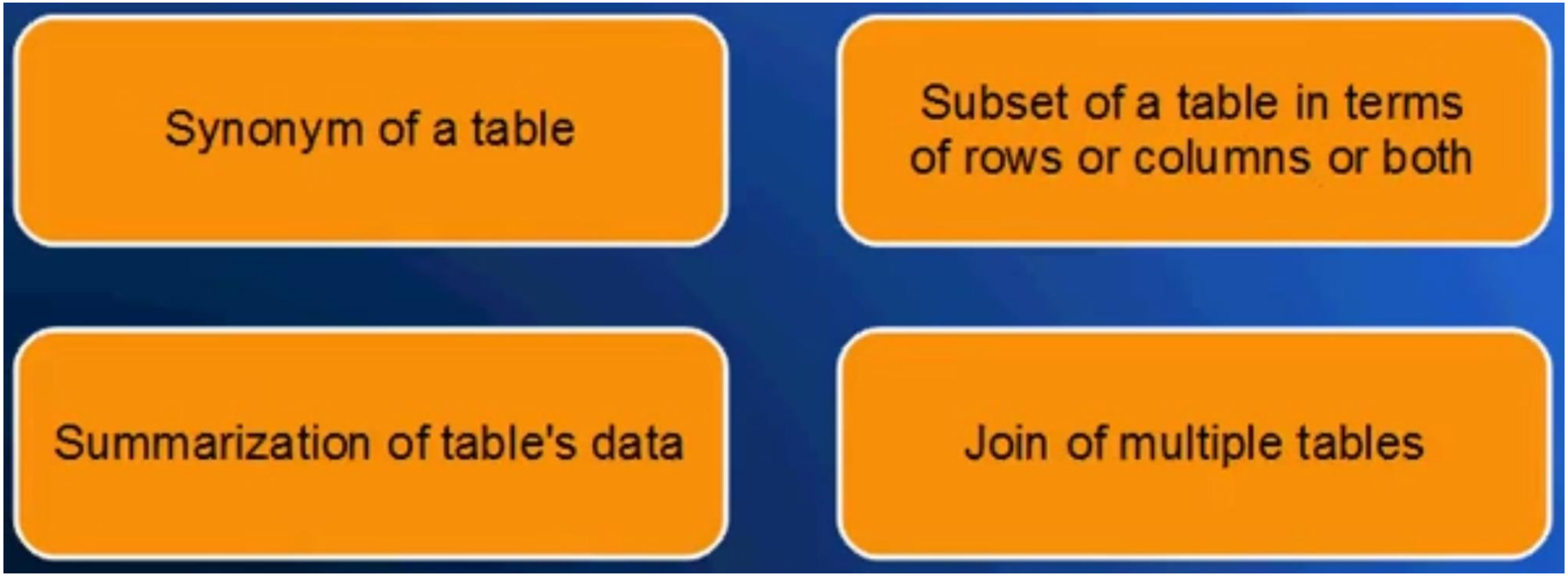
View: Emp_10

ID	NAME
1	Hasan
2	Bill
4	Steve

Views



View Types Based on Purpose



Synonym of a table

Subset of a table in terms of rows or columns or both

Summarization of table's data

Join of multiple tables

Views - Benefits

- Reduce complexity
- Provide a level of security
- Provide a mechanism to customize the appearance of the database
- Present a consistent, unchanging picture of the structure of the database, even if the underlying database is changed

Updating Views

- All updates to a base relation should be immediately reflected in all views that reference that base relation.

- If view is updated, underlying base relation should reflect change.

Updating Views

- There are restrictions on types of modifications that can be made through views:
  - Updates are allowed if query involves a single base relation and contains a candidate key of base relation.
  - Updates are not allowed involving multiple base relations.
  - Updates are not allowed involving aggregation or grouping operations.
- You will learn more about Views in SQL lecture

Outline of ERD

- Conceptual data modeling
  - ERD: entity-relationship diagram
- Notations
  - Entities, relationships, cardinalities
- Relationships
  - One-to-one, one-to-many, many-to-many
  - Identifying Relationship

ER Diagram Terminologies

- Entities, attributes, identifiers, and relationships
- Entity:
  - Thing or object, whether real or imagined, about which information needs to be known/tracked
  - Examples:
    - Customer
    - Order
    - Student
    - Book
    - Employee

Entity

- Entity class/type:
  - A collection that share common properties or characteristics
  - UPPER CASE (noun)
- Entity instance:
  - Representation of a particular entity
- Many instances in an entity class

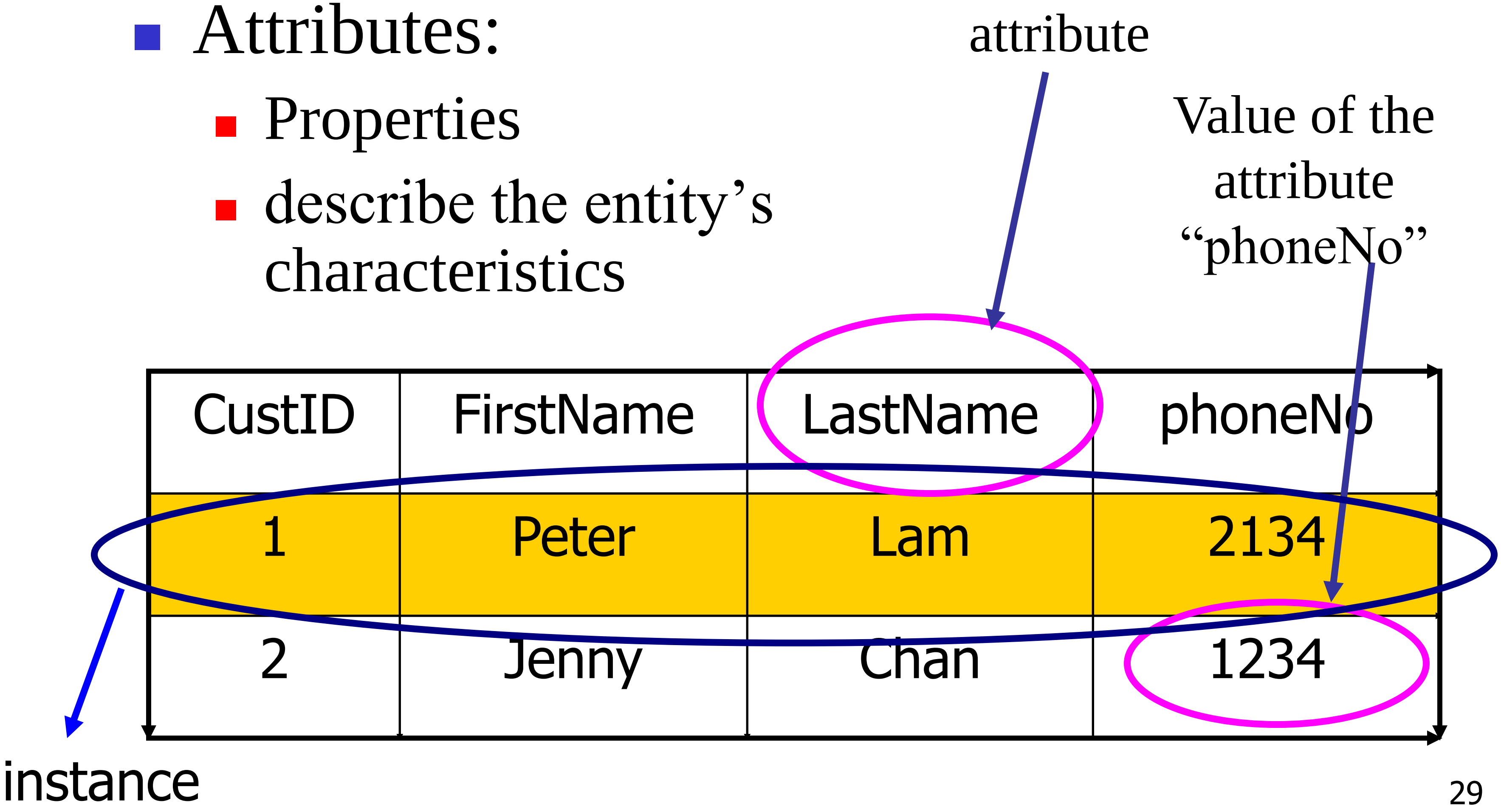
CUSTOMER entity class contains:
CustID
FirstName
LastName
phoneNo

Two instances:

1	Peter	Lam	2134
2	Jenny	Chan	1234

Attributes

- Attributes:
  - Properties
  - describe the entity’s characteristics



CustID	FirstName	LastName	phoneNo
1	Peter	Lam	2134
2	Jenny	Chan	1234

attribute

Value of the attribute “phoneNo”

instance

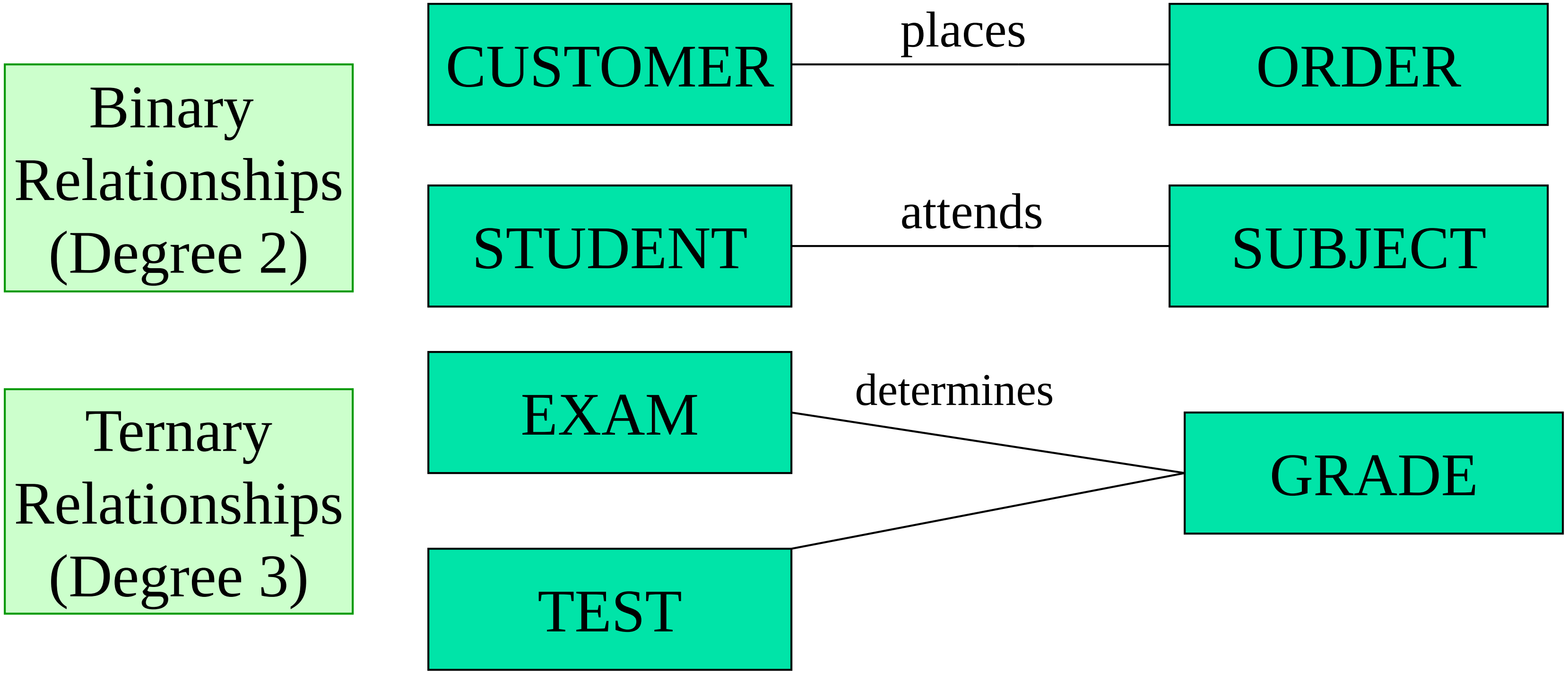
Identifiers

- Primary key
- Support entity identification
- One or more of the entity's attribute that uniquely identify a row in the table
  - CustID: uniquely identify a row
  - Not likely identified by LastName
  - Composite identifiers: identifier with 2 or more attributes

CustID	FirstName	LastName	phoneNo
1	Peter	Lam	2134
2	Jenny	Chan	1234

Relationships

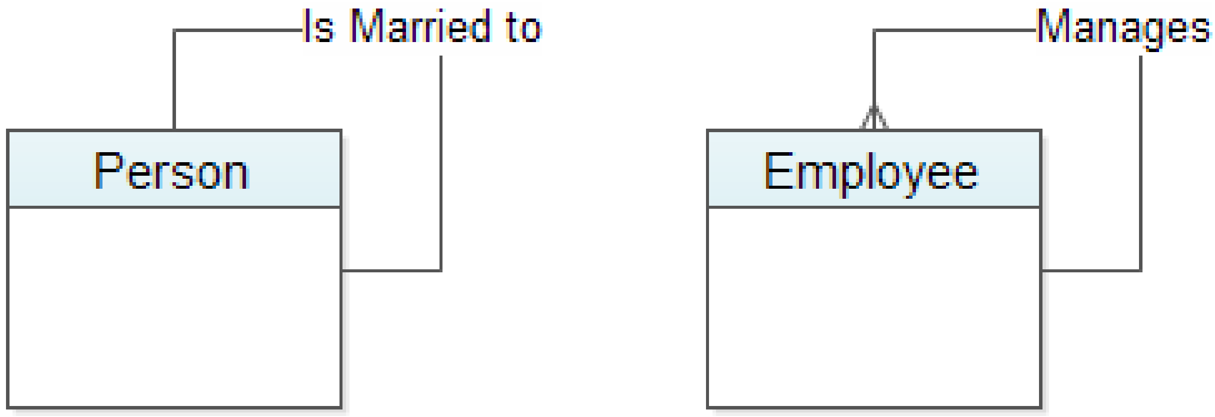
- Describes how one or more entities are related to each other
- Entity: noun; Relationship: verb


Binary Relationships (Degree 2)
CUSTOMER
places
ORDER
STUDENT
attends
SUBJECT
Ternary Relationships (Degree 3)
EXAM
determines
GRADE
TEST

Relationships

- Describes how one or more entities are related to each other
- Entity: noun; Relationship: verb

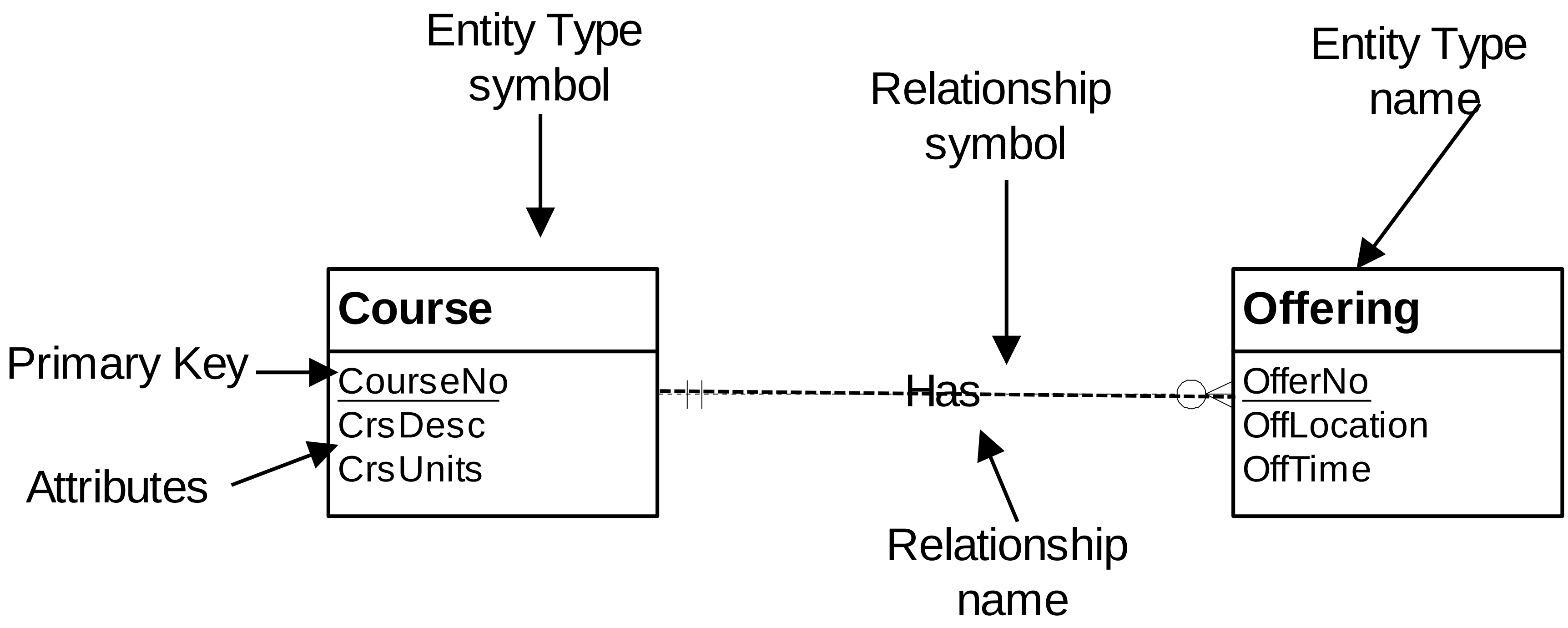
Unary Relationships (Degree 1)



One-to-One

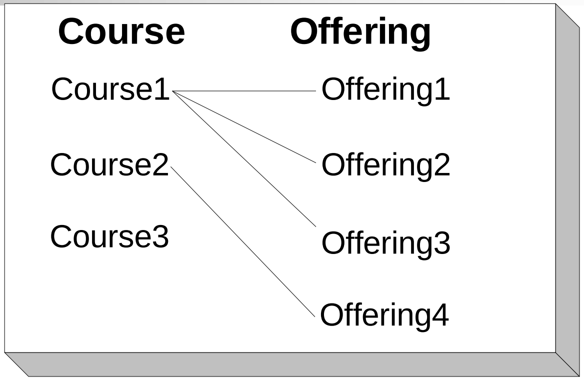
One-to-Many

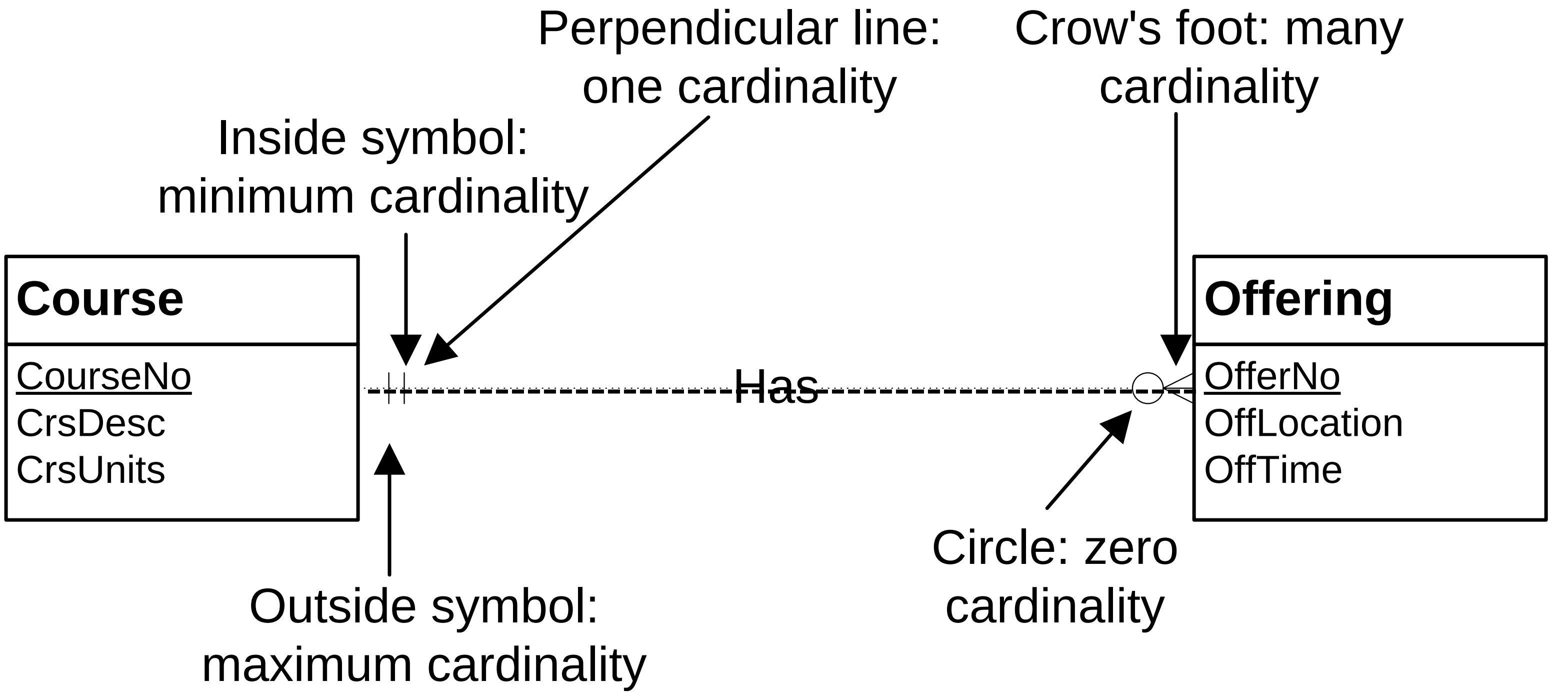
ER Diagrams


Entity Type symbol
Relationship symbol
Entity Type name
Primary Key
Attributes
Course
CourseNo
CrsDesc
CrsUnits
Has
Offering
OfferNo
OffLocation
OffTime
Relationship name

Cardinalities

- Contains the number of objects that participate in a relationship





Perpendicular line: one cardinality

Crow's foot: many cardinality

Inside symbol: minimum cardinality

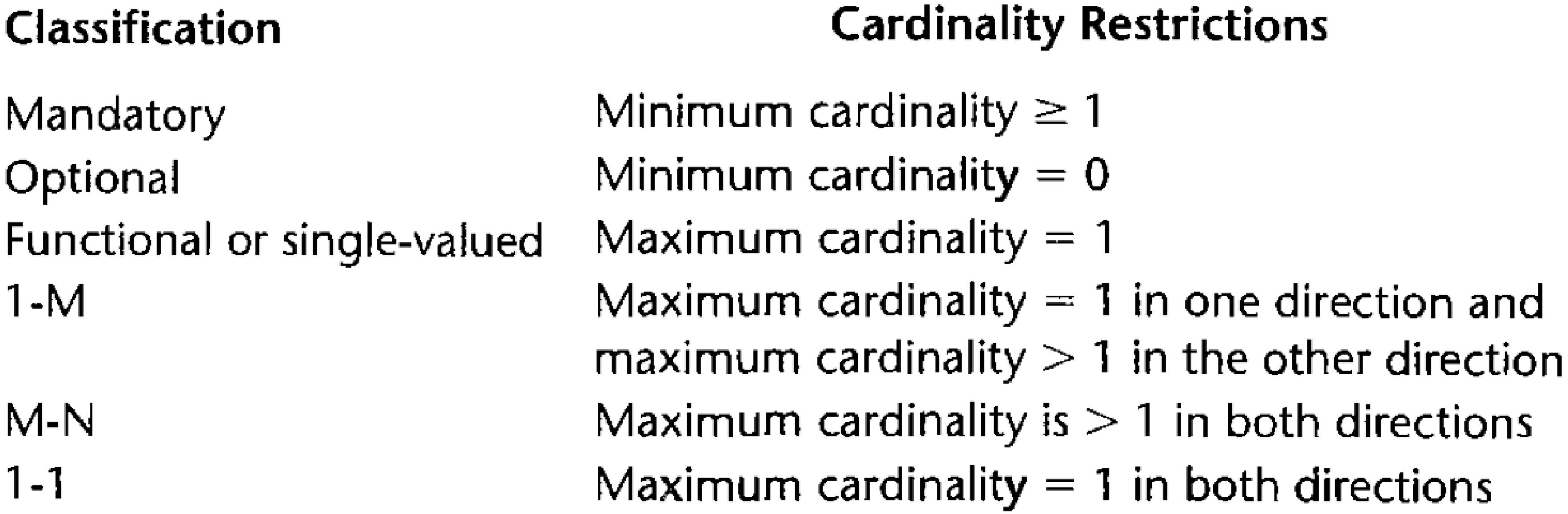
Outside symbol: maximum cardinality

Circle: zero cardinality

Cardinalities

- Minimum cardinality:
  - Indicates whether participation in the relationship is
    - Mandatory (1)
    - Optional (0)
- Maximum cardinality:
  - The maximum number of entities that can occur on one side of the relationship
    - One-to-one
    - One-to-many
    - Many-to-many

Cardinalities

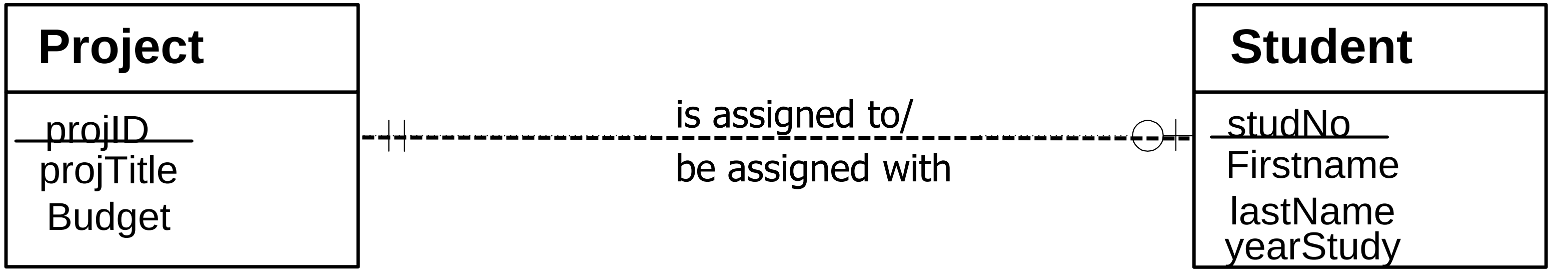


Classification	Cardinality Restrictions
Mandatory	Minimum cardinality ≥ 1
Optional	Minimum cardinality = 0
Functional or single-valued	Maximum cardinality = 1
1-M	Maximum cardinality = 1 in one direction and maximum cardinality > 1 in the other direction
M-N	Maximum cardinality is > 1 in both directions
1-1	Maximum cardinality = 1 in both directions

Example 1

- Project:
  - Project title (ProjTitle), Project ID (ProjID) and Budget (Budget)
- Student:
  - Year of study (YearStudy), Student Number (StudNo), First Name (FirstName) and Last Name (LastName)
- Relationship:
  - A project might be assigned to at most one student
  - A student must be assigned one project

Example 1: answer

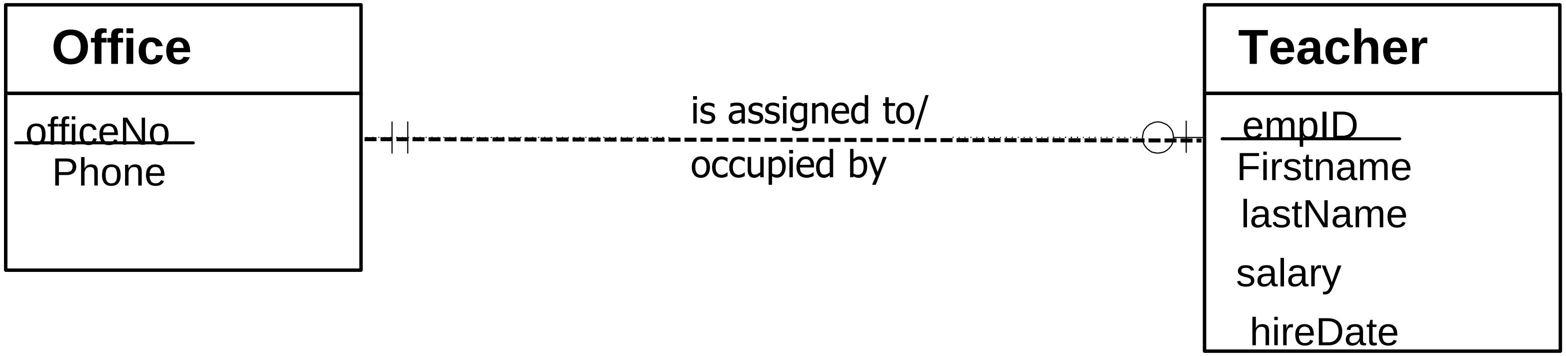


- A project might be assigned to at most one student
- A student must be assigned one project

Example 2

- Teacher:
  - Employee ID (ID), salary, firstname, lastname, hireDate
- Office:
  - officeNo, phone
- Each teacher is assigned to one office and an office is occupied by at most one teacher

Example 2: Answer



Office

officeNo
Phone

is assigned to/
occupied by

Teacher

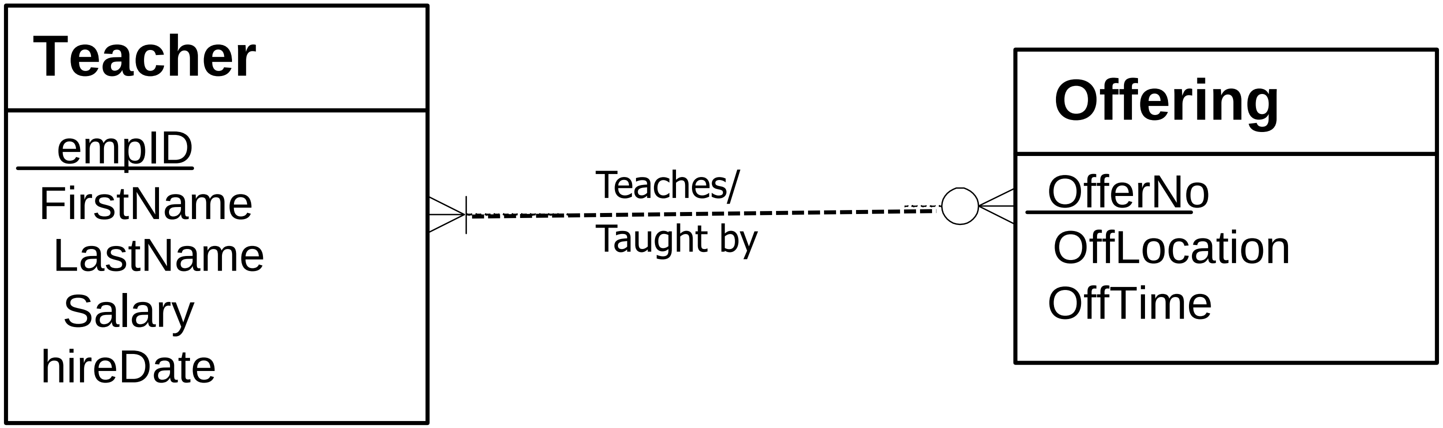
empID
Firstname
lastName
salary
hireDate

- Each teacher is assigned to one office and an office is occupied by at most one teacher

Example 3

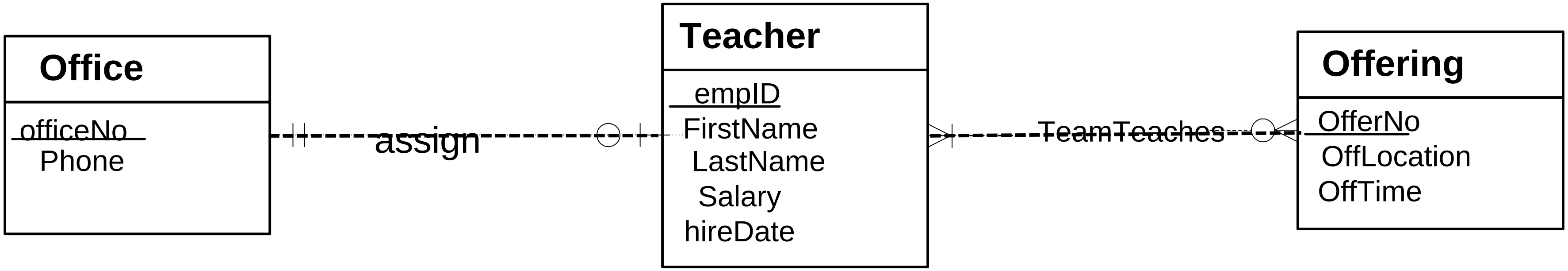
- Teacher:
  - Employee ID (ID), salary, firstname, lastname, hireDate
- Offering:
  - offerNo, Location, Time
- Multiple teachers can jointly teach one offering
- A teacher may or may not teach any offering, but a teacher can also teach many offerings
- One offering must be taught by at least one teacher

Example 3: Answer



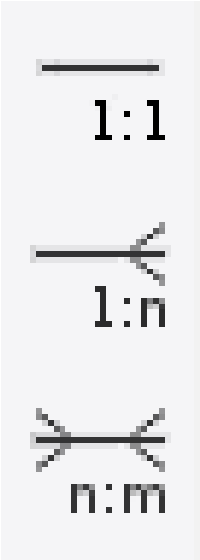
- Multiple teachers can jointly teach one offering
- A teacher may or may not teach any offering, a teacher can also teach many offerings
- one offering must be taught by at least one teacher

Example 2 + Example 3

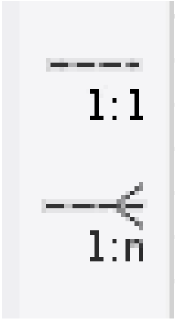


In MySQL Workbench …

- An ***identifying relationship***: identified by a solid line between tables
  - An identifying relationship is one where the child table cannot be uniquely identified without its parent. Typically this occurs where an intermediary table is created to resolve a many-to-many relationship. In such cases, the primary key is usually a composite key made up of the primary keys from the two original tables.

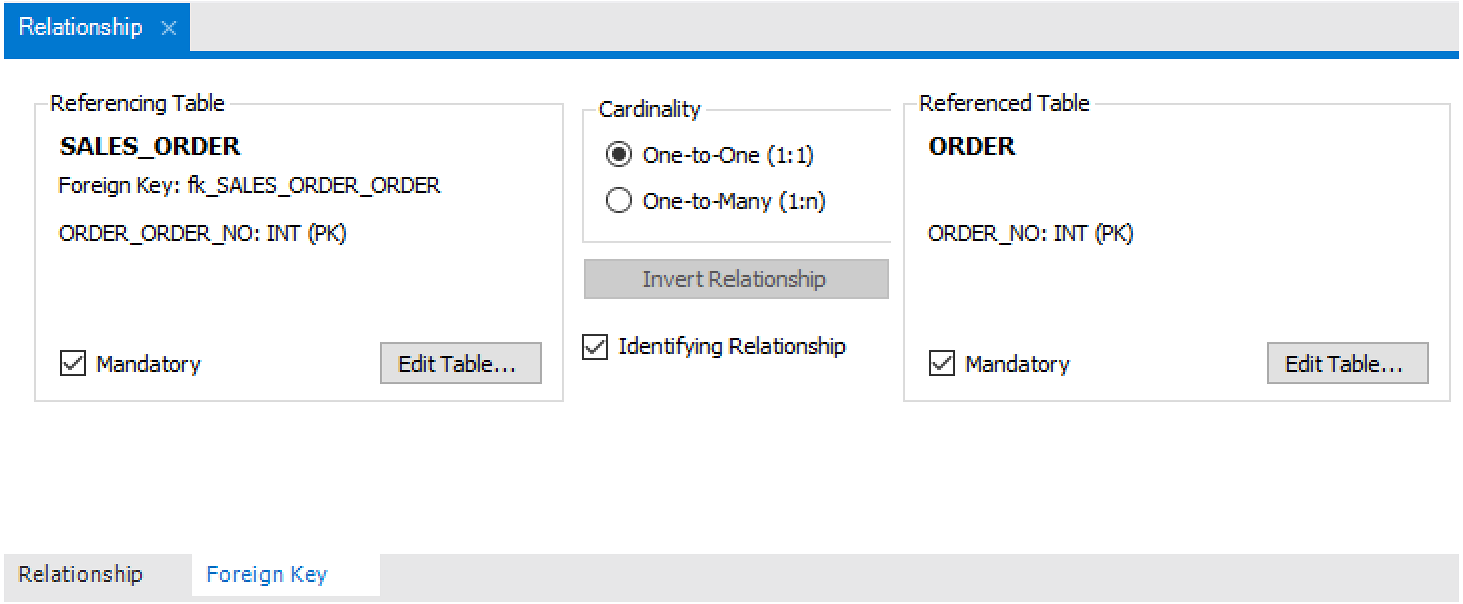


- A ***non-identifying relationship***: identified by a broken (dashed) line between tables



In MySQL Workbench …

- To edit relationship in MySQL Workbench,
  - Double click the relationship between tables, or
  - Right click the relationship between tables and select “Edit Relationship”



Referencing Table

SALES_ORDER

Foreign Key: fk_SALES_ORDER_ORDER

ORDER_ORDER_NO: INT (PK)

☑ Mandatory

Cardinality

● One-to-One (1:1)

○ One-to-Many (1:n)

☑ Identifying Relationship

Referenced Table

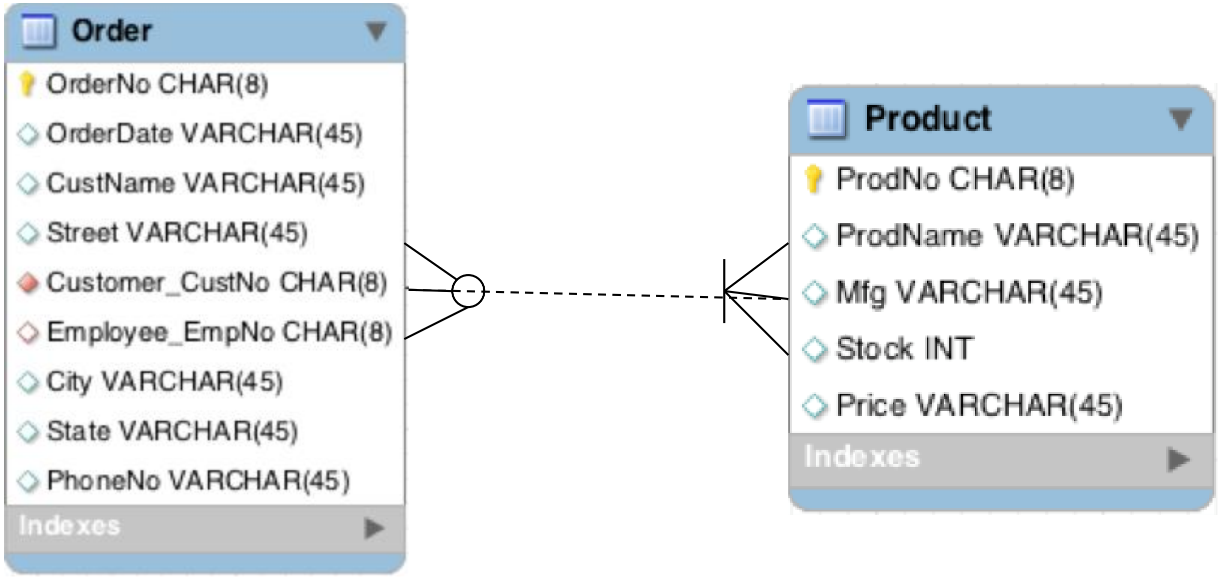
ORDER

ORDER_NO: INT (PK)

☑ Mandatory

Implement M-N Relationships

- An *order* is associated with a number of *products* and a *product* can be associated with many *orders*.
- Each order should have at least one product. But a product may not have any order.
- The relationship between Order and Product is M:N

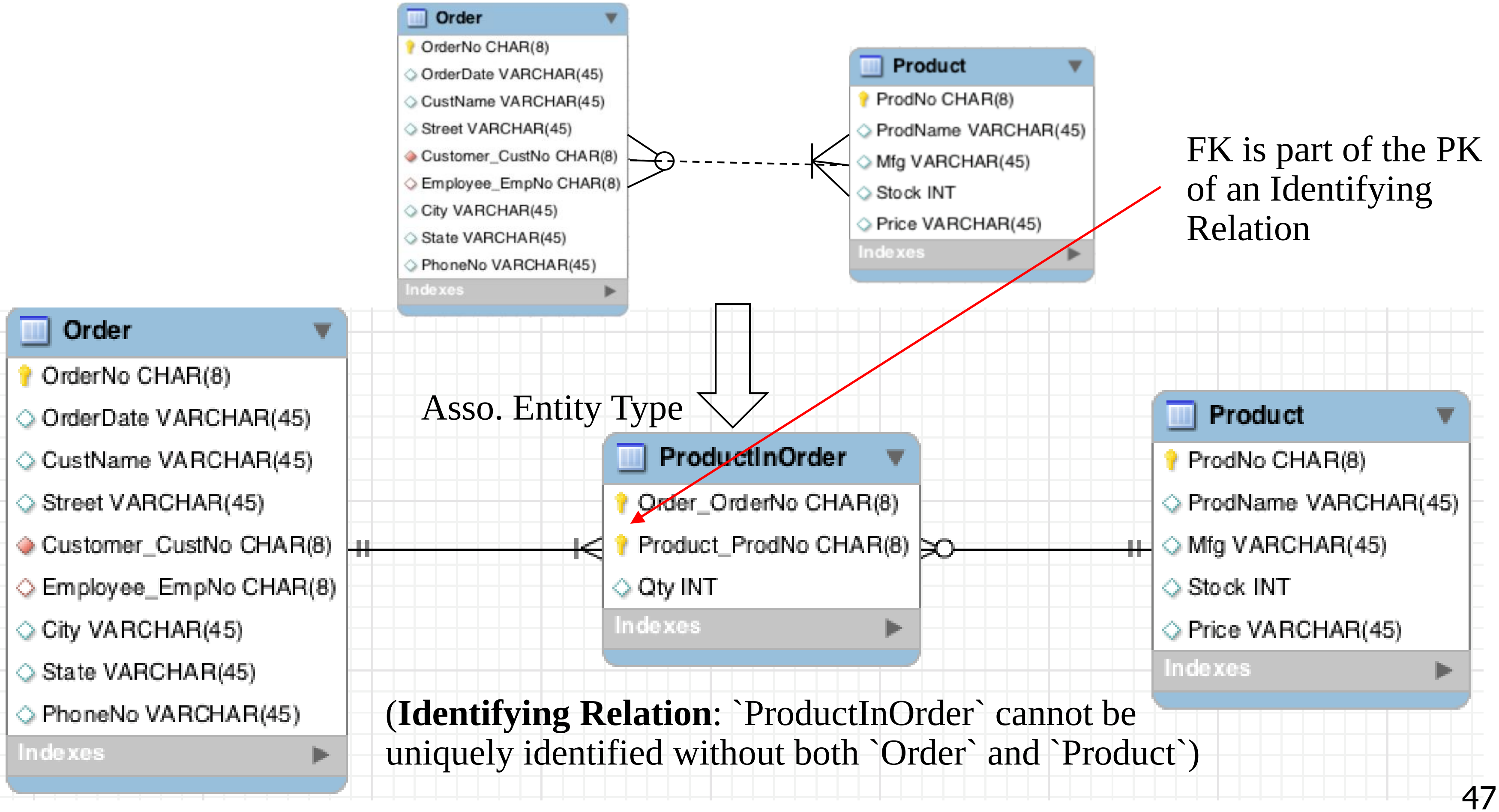


Order
OrderNo CHAR(8)
OrderDate VARCHAR(45)
CustName VARCHAR(45)
Street VARCHAR(45)
Customer_CustNo CHAR(8)
Employee_EmpNo CHAR(8)
City VARCHAR(45)
State VARCHAR(45)
PhoneNo VARCHAR(45)
Indexes

Product
ProdNo CHAR(8)
ProdName VARCHAR(45)
Mfg VARCHAR(45)
Stock INT
Price VARCHAR(45)
Indexes

Implement M-N Relationships

- Transform M:N to 1:M Relationship:
  - M:N = two 1-M + associative entity type



Order
- OrderNo CHAR(8)
- OrderDate VARCHAR(45)
- CustName VARCHAR(45)
- Street VARCHAR(45)
- Customer_CustNo CHAR(8)
- Employee_EmpNo CHAR(8)
- City VARCHAR(45)
- State VARCHAR(45)
- PhoneNo VARCHAR(45)

Indexes

Product
- ProdNo CHAR(8)
- ProdName VARCHAR(45)
- Mfg VARCHAR(45)
- Stock INT
- Price VARCHAR(45)

Indexes

FK is part of the PK of an Identifying Relation

Order
- OrderNo CHAR(8)
- OrderDate VARCHAR(45)
- CustName VARCHAR(45)
- Street VARCHAR(45)
- Customer_CustNo CHAR(8)
- Employee_EmpNo CHAR(8)
- City VARCHAR(45)
- State VARCHAR(45)
- PhoneNo VARCHAR(45)

Indexes

Asso. Entity Type

ProductInOrder
- Order_OrderNo CHAR(8)
- Product_ProdNo CHAR(8)
- Qty INT

Indexes

Product
- ProdNo CHAR(8)
- ProdName VARCHAR(45)
- Mfg VARCHAR(45)
- Stock INT
- Price VARCHAR(45)

Indexes

(**Identifying Relation**: `ProductInOrder` cannot be uniquely identified without both `Order` and `Product`)

Implement M-N Relationships

Tutorial Q2(a)

- A database involves the relationship between ***parts***, the ***vendors*** providing the parts, and the ***products*** that use the parts. Specifically, a part is supplied by at least one vendor. Each vendor may supply many parts; but some new vendors may supply none. Each product is composed of one or more parts. A part may be used by many products; but some new parts may not be used by any product. Draw an entity-relationship diagram to illustrate the relationships among the entities Part, Vendor, and Product. In your solution, many-to-many relationships should be converted into one-to-many relationships. It is not necessary to provide attributes for the entities.