

SQL FOR DB2

Chapter 6

Entity Relationship Diagrams

CHAPTER OBJECTIVES

- ① Explain what entity relationship diagrams(ERDs) are
- ① Explain advantages and disadvantages of ERDs
- ① Identify and explain the steps for developing an ERD

BASIC ERD CONCEPTS AND SYMBOLS

- ◎ An entity relationship diagram (ERD) is a diagram model used to organize the data in a project into entities and to define the relationships between the entities
- ◎ In ERDs, rectangles represent entities, and lines connecting the rectangles represent relationships hence the name “entity relationship”
- ◎ ERDs contain:
 - ◎ Entities
 - ◎ Relationships
 - ◎ Attributes
 - ◎ Cardinality
 - ◎ Primary keys and Foreign keys

ENTITY

- ⊙ A data entity is anything real or abstract about which the company wants to store data
- ⊙ Entities are like nouns consisting of persons, places or things, such as students, courses, rooms and professors

Department	

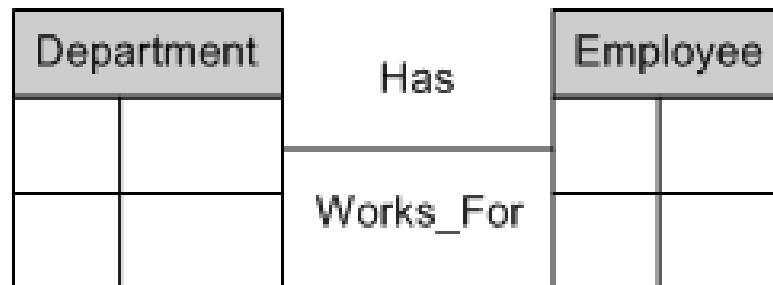
Supervisor	

Employee	

Project	

RELATIONSHIPS

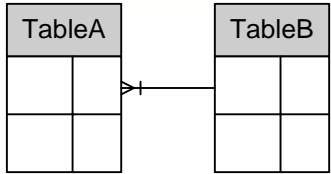
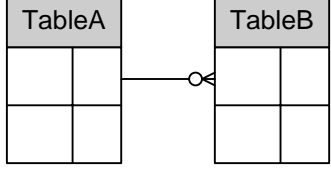
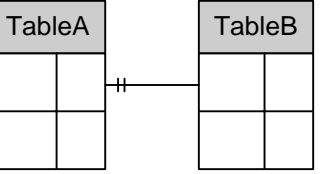
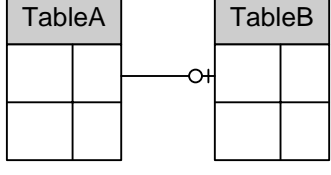
- ⦿ A relationship is like a verb that show some dependency or natural association between two or more entities
- ⦿ Relationships between entities are represented by lines joining the two entities
- ⦿ A relationship line can be labeled to define the relationship



OPTIONALITY AND CARDINALITY

- ◎ Symbols are placed at the ends of relationship lines to indicate the optionality and the cardinality of each relationship
- ◎ Optionality expresses whether the relationship is optional or mandatory
- ◎ Cardinality defines the number of occurrences on one entity for a single occurrence of the related entity
- ◎ Two symbols are specified at the end of the relationship line.
- ◎ The first symbol is the optionality symbol:
 - ◎ Circle is option
 - ◎ A line is mandatory
- ◎ The second symbol is the cardinality symbol:
 - ◎ A stroke indicates 1
 - ◎ A “crow’s foot” indicates many

ERD NOTATION AND MEANING

	<p>Each instance of TableA is related to a minimum of zero and a maximum of one instance of TableB</p>
	<p>Each instance of TableB is related to a minimum of one and a maximum of one instance of TableB</p>
	<p>Each instance of TableA is related to a minimum of zero and a maximum of many instances of TableB</p>
	<p>Each instance of TableB is related to a minimum of one and a maximum of many instances of TableB</p>

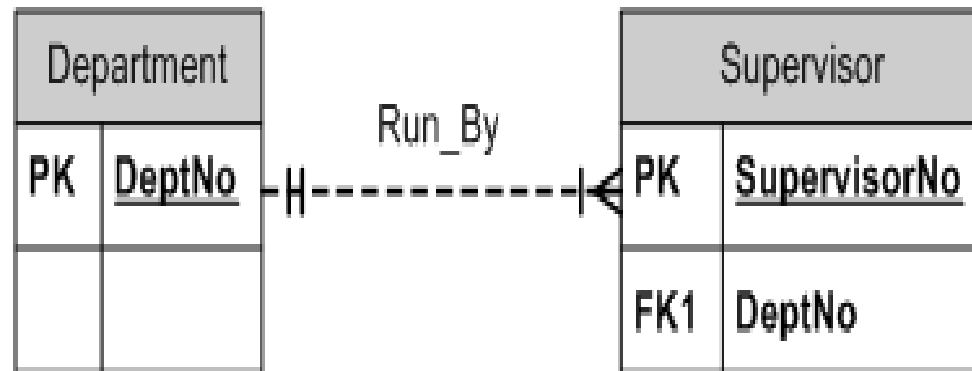
ATTRIBUTE

- ⊙ An attribute is some identifying or descriptive piece of information for an entity
- ⊙ Attributes are like adjectives that describe an entity
- ⊙ An attribute or combination of attributes that uniquely identifies one and only one instance of an entity is called a primary key.
- ⊙ The primary key is underlined and identified by the letters PK

Employee	
PK	<u>EmployeeNo</u>
	FirstName LastName HireDate Dependents

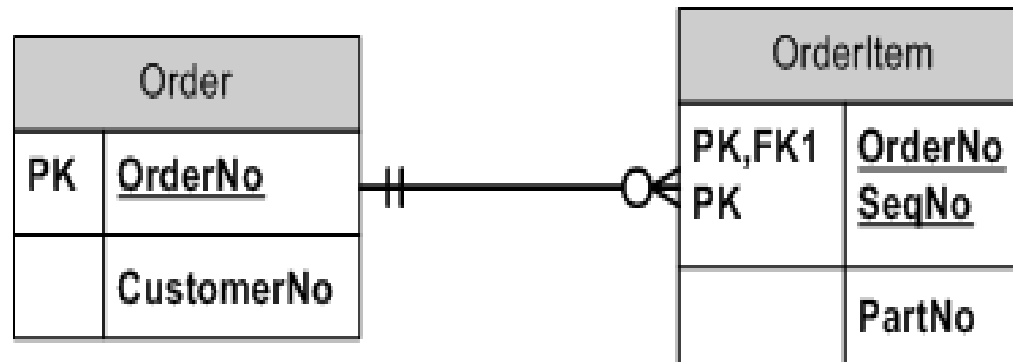
FOREIGN KEY

- ⊙ When a relationship exists between two entities, the primary key of one entity becomes a foreign key in the other entity
- ⊙ The letters FK are placed beside the foreign key



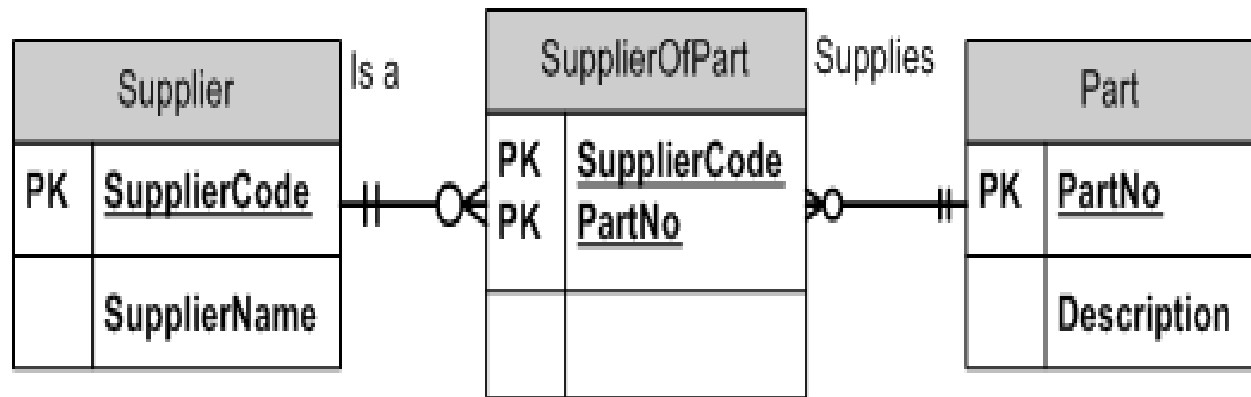
DEPENDENT TABLE OR WEAK ENTITY

- ⊙ A weak entity is one for which an instance cannot exist unless a corresponding entity on the other end of the relationship exists
- ⊙ An entity that is not a weak entity is known as a regular entity



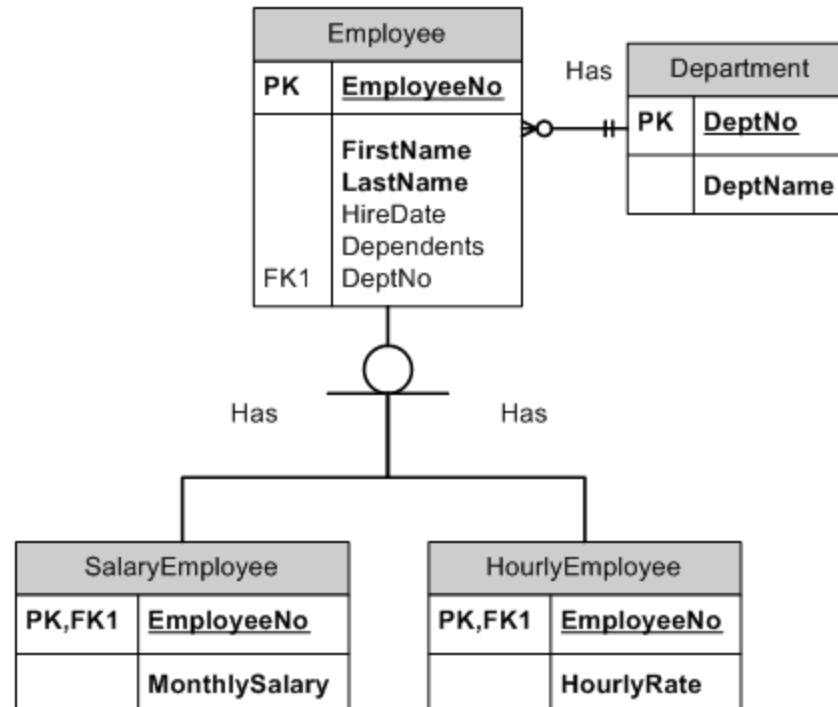
ASSOCIATIVE ENTITIES

- ⊙ For many-to-many relationships, create a new table
- ⊙ This is called an **associative entity** because it is created from the association between two entities.
- ⊙ The key to the new entity is the combination of the keys from its source entities, this is called a concatenated key
- ⊙ Each of the attributes in the new table which compose the concatenated key in the new table are also called **foreign keys** because they are keys to another table



ENTITY SUBTYPES

- ⊙ When two potential entities are encountered that seem very similar but not quite the same, ERD entity subtypes can be used to represent them



DEVELOPING ENTITY RELATIONSHIP DIAGRAMS

Step	Title	Description
1	Identify Entities	Identify the people, places or things about which the end-users want to store data.
2	Determine Relationships	Find the natural associations between pairs of entities using a relationship matrix. Arrange entities in rectangles and join those entities with a line.
3	Determine Optionality and Cardinality	Determine the number of occurrences of one entity for a single occurrence of the related entity.
4	Resolve Many-to-Many Relationships	Many-to-many relationships cannot be implemented into database tables because each row will need an indefinite number of attributes to maintain the many-to-many relationship. Many-to-many relationships must be converted to associative entities.
5	Identify Primary Keys	Identify the data attribute(s) that uniquely identify one and only one occurrence of each entity. Eliminate Many-to-Many relationships and include primary and foreign keys in each entity.
6	Identify Attributes	Identify the attributes which are essential to the system under development. For each attribute, match it with exactly one entity that it describes.
7	Determine Data Types	Identify the data types and sizes of each attribute.
8	Check Results	Examine the ERD from the point of view of a system owner or user.

CHAPTER SUMMARY

- ◎ Basic ERD Concepts
 - ◎ Entity – like a noun consisting of persons, places, or things.
 - ◎ Relationship – like a verb that shows some dependency between two or more entities
 - ◎ Optionality – expresses whether the relationship is optional or mandatory
 - ◎ Cardinality – defines the number of occurrences of one entity for a single occurrence of the related entity.
 - ◎ Primary key – uniquely identifies each row in an entity.
 - ◎ Foreign key – links the related rows between two entities.
 - ◎ Attribute – like adjectives that describe an entity.

CHAPTER SUMMARY

- ⊙ ERD diagrams
 - ⊙ Represent entities, relationships, and their attributes in pictorial form
 - ⊙ Use rectangles to represent entities
 - ⊙ Use lines to represent relationships
 - ⊙ Attributes are listed in the rectangles with annotation to show required and optional values and identifier attributes and referencing attributes.
 - ⊙ Can show alternative types of relationships, including one-to-one, one-to-many, or many-to-many relationships
 - ⊙ Shows whether one or more related instances are optional or required