

Database System

ER Diagram

ผศ.ดร.พรเทพ ไรจนวสุ

สาขาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีสารสนเทศและการสื่อสาร

วัตถุประสงค์การเรียนรู้

1. ให้คำนิยามคำศัพท์ business rule, E-R model, E-R diagram, entity แบบต่าง ๆ , attribute แบบต่าง ๆ , relationship แบบต่าง ๆ
2. สามารถบอกขั้นตอนการออกแบบฐานข้อมูลได้
3. สามารถกำหนด entity attributes และความสัมพันธ์ได้
4. สามารถสร้าง E-R diagram จากสถานการณ์ที่กำหนดได้

ทำไมต้องมีการออกแบบฐานข้อมูล

Why are there blanks in rows 9 and 10?

How to produce an alphabetical listing of employees?

How to count how many employees are certified in Basic Database Manipulation?

Is Basic Database Manipulation the same as Basic DB Manipulation?

What if an employee acquires a fourth certification?
Do we add another column?

ID	ENum	Name	Title	HireDate	Skill1	Skill1Date	Skill2	Skill2Date	Skill3	Skill3Date
1	02345	Brian Oates	DBA	2/14/1995	Basic Database Management	2/14/2002	Advanced Database Management	2/14/2005	Basic Web Design	8/9/2003
2	08273	Marco Bienz	Analyst	7/28/2006	Basic Web Design	3/8/2009	Advance Process Modeling	8/19/2012		
3	06234	Jasmine Patel	Programmer	8/10/2005	Basic Web Design	8/10/2007	Advanced C# programming	8/10/2007	Basic DB manipulation	1/29/2012
4	03373	Franklin Johnson, Jr.	Purchasing Agent	3/15/2002	Advanced Spreadsheets	6/20/2011				
5	13567	Almond, Robert	Analyst	9/30/2012	Basic Process Modeling	9/30/2014	Basic Database Design	5/23/2015		
6	10282	Richardson, Amanda	Clerk	4/11/2011						
7	09382	Susan Mathis	Database Programmer	8/2/2010	Basic DB Design	8/2/2012	Basic Database Manipulation	8/2/2012	Advanced DB Manipulation	5/1/2013
8	14311	Duong, Lee	Programmer	9/1/2014	Basic Web Design	9/1/2016				
9					Master Database Programming					
10					Basic Spreadsheets					
11	09002	Wade Gaither	Clerk	5/20/2010	Advanced Spreadsheets	5/16/2013	Basic Web Design	5/16/2013		
12	13383	Raymond F. Matthews	Programmer	3/12/2012	Basic C# Programming	3/12/2014				
13	09283	Chavez, Juan	Clerk	7/4/2010						
14	04893	Patricia Richards	DBA	6/11/2004	Advanced Database Management	6/11/2006	Advanced Database Manipulation	9/20/2012		
15	13832	Lee, Megan	Programmer	9/29/2013						

Database System

Table name: EMPLOYEE

Employee_ID	Employee_FName	Employee_LName	Employee_HireDate	Employee_Title
02345	Johnny	Jones	2/14/1995	DBA
03373	Franklin	Johnson	3/15/2002	Purchasing Agent
04893	Patricia	Richards	6/11/2004	DBA
06234	Jasmine	Patel	8/10/2005	Programmer
08273	Marco	Bienz	7/28/2006	Analyst
09002	Ben	Joiner	5/20/2010	Clerk
09283	Juan	Chavez	7/4/2010	Clerk
09382	Jessica	Johnson	8/2/2010	Database Programmer
10282	Amanda	Richardson	4/11/2011	Clerk
13383	Raymond	Matthews	3/12/2012	Programmer
13567	Robert	Almond	9/30/2012	Analyst
13932	Megan	Lee	9/29/2013	Programmer
14311	Lee	Duong	9/1/2014	Programmer

Database name: Ch01_Text

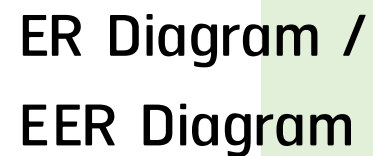
Table name: CERTIFIED

Employee_ID	Skill_ID	Certified_Date
02345	100	2/14/2002
02345	110	8/9/2003
02345	180	2/14/2006
03373	120	6/20/2011
04893	180	6/11/2006
04893	220	9/20/2012
06234	110	8/10/2007
06234	200	8/10/2007
06234	210	1/29/2012
08273	110	3/8/2009
08273	190	8/19/2012
09002	110	5/16/2013
09002	120	5/16/2013
09382	140	8/2/2012
09382	210	8/2/2012
09382	220	5/1/2013
13383	170	3/12/2014
13567	130	9/30/2014
13567	140	5/23/2015
14311	110	9/1/2016

Table name: SKILL

Skill_ID	Skill_Name	Skill_Description
100	Basic Database Management	Create and manage database user accounts.
110	Basic Web Design	Create and maintain HTML and CSS documents.
120	Advanced Spreadsheets	Use of advanced functions, user-defined functions, and macroing.
130	Basic Process Modeling	Create core business process models using standard libraries.
140	Basic Database Design	Create simple data models.
150	Master Database Programming	Create integrated trigger and procedure packages for a distributed environment.
160	Basic Spreadsheets	Create single tab worksheets with basic formulas
170	Basic C# Programming	Create single-tier data aware modules.
180	Advanced Database Management	Manage Database Server Clusters.
190	Advance Process Modeling	Evaluate and Redesign cross-functional internal and external business processes.
200	Advanced C# Programming	Create multi-tier applications using multi-threading
210	Basic Database Manipulation	Create simple data retrieval and manipulation statements in SQL
220	Advanced Database Manipulation	Use of advanced data manipulation methods for multi-table inserts, set operations, and correlated subqueries.

Data Flow Diagram / Sequence Diagram



เก็บความต้องการของผู้ใช้งาน

- Data Requirement
- Functional Requirement

Figure 3.1

A simplified diagram to illustrate the main phases of database design.

ขั้นตอนการออกแบบฐานข้อมูล

1. นักออกแบบฐานข้อมูลเก็บความต้องการของผู้ใช้งานระบบและสอบถามจากผู้เชี่ยวชาญ
2. เลือกแบบจำลองข้อมูลเพื่อสร้างเป็นแบบแผนแนวคิด (conceptual schema) ของฐานข้อมูล
3. ตรวจสอบแบบแผนแนวคิดให้สอดคล้องกับความต้องการการทำงาน
4. แปลงแบบแผนแนวคิดสู่การใช้งานจริง เริ่มจากการแปลงแบบแผนแนวคิดสู่การออกแบบเชิงตรรก (logical design) คือแบบแผนอีอาร์สู่แบบแผนฐานข้อมูล
5. หลังจากนั้นเป็นการออกแบบระดับกายภาพ (physical design) เช่นการกำหนดโครงสร้างข้อมูลของระบบฐานข้อมูลและการกำหนดดัชนี

E-R Model

- เป็นแนวคิดในการเก็บข้อมูล (Data model) ระดับ conceptual
- ใช้สำหรับออกแบบฐานข้อมูลสำหรับโปรแกรมระบบฐานข้อมูลต่าง ๆ
- E-R Diagram เป็นแผนภาพแสดง E-R Model

ตัวอย่างระบบร้านเช่าหนังสือ

1 ระบบหนังสือ

- - สามารถเพิ่ม/ลบ/แก้ไขข้อมูลหนังสือในระบบได้
- - สามารถค้นหาหนังสือจากชื่อหนังสือได้
- - สามารถค้นหาหนังสือจากหมายเลขหนังสือ/ISBNได้
- - สามารถตรวจสอบสถานะของหนังสือในระบบได้

2 ระบบการเช่า-คืนหนังสือ

- - สามารถทำการเช่าและบอกรายละเอียดข้อมูลของการเช่าได้
- - สามารถทำการคืนและบอกรายละเอียดข้อมูลของการคืนได้
- - สามารถตรวจสอบรายการเช่า-คืนหนังสือของสมาชิกได้
- - สามารถคำนวณราคาค่าเช่าหนังสือได้
- - สามารถคำนวณราคาค่าปรับของหนังสือที่ส่งคืนล่าช้าได้
- - สามารถตรวจสอบยอดค่าส่งคืนหนังสือได้
- - สามารถออกใบเสร็จการชำระเงินได้

3 ระบบสมาชิก

- - สามารถเพิ่ม/ลบ/แก้ไขข้อมูลสมาชิกในระบบได้
- - สามารถค้นหารายละเอียดลูกค้าจากรหัสสมาชิกได้
- - สามารถค้นหารายละเอียดลูกค้าจากชื่อสมาชิกได้
- - สามารถออกบัตรสมาชิกให้แก่ผู้ใช้บริการได้

4 ระบบรายงาน

- - สามารถออกรายงานการเช่า-คืนได้
- - สามารถออกรายงานผู้มาเยี่ยมได้
- - สามารถออกรายงานหนังสือค้างส่งคืนได้
- - สามารถออกรายงานหนังสือทั้งหมดได้

ต้องเก็บข้อมูลอะไรบ้าง

System Database

หนังสือ

สมาชิก

ข้อมูลการยืม-คืน ?

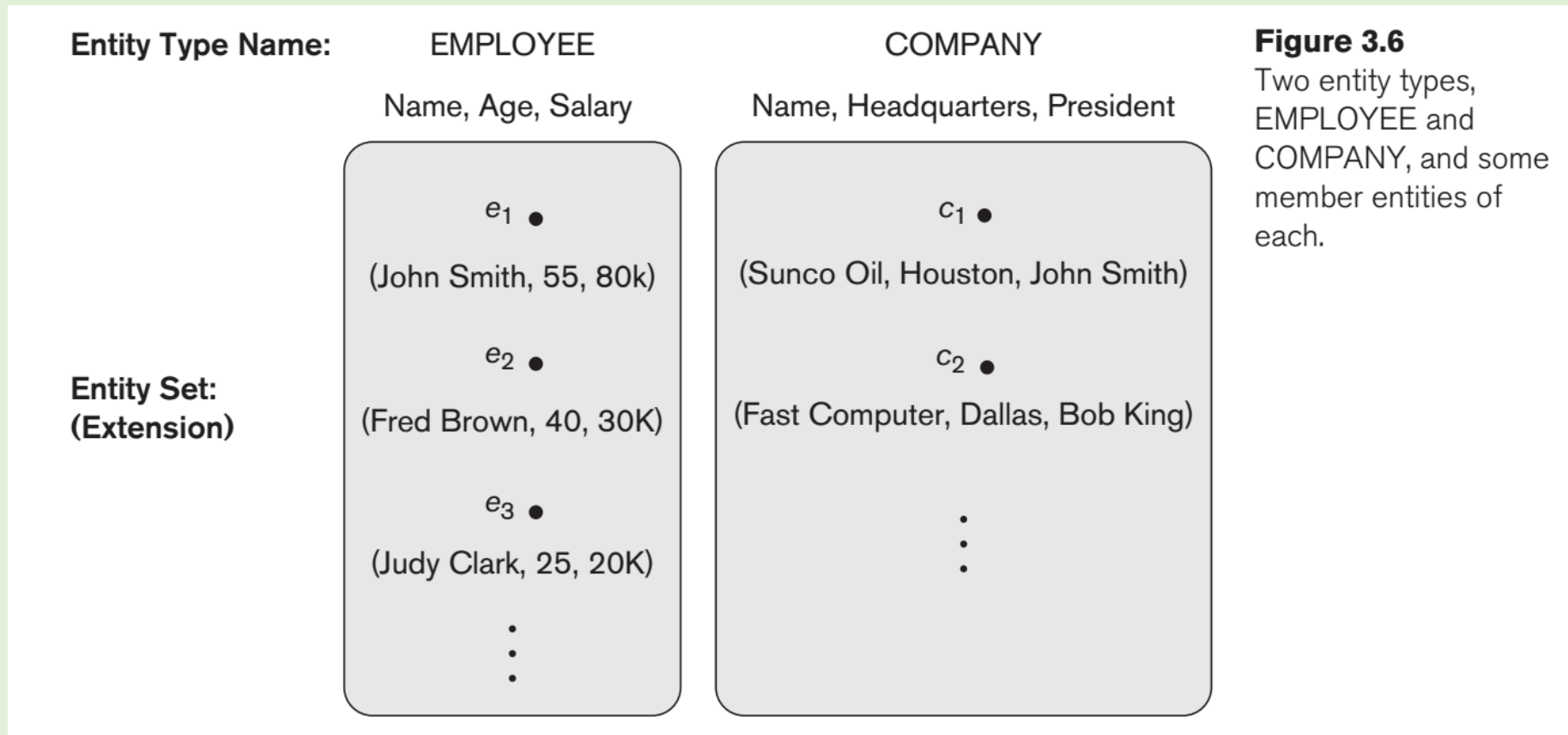
Entity, Relationship, Attributes

As the name implies, an ER diagram models data as **entities** and **relationships**, and entities have **attributes**. An **entity** is a thing about which we store data, for example, a person, a bank account, a building. In the original presentation, Chen (1976) described an entity as a "thing which can be distinctly identified." So an entity can be a person, place, object, event, or concept about which we wish to store data.

Database Design Using Entity-Relationship Diagrams, Auerbach Publications , 2003

- เป็นวัตถุที่มีตัวตน เช่น รถ บ้าน พนักงาน สินค้า
- เป็นวัตถุนามธรรม เช่น รายวิชา อาชีพ บริษัท ทริปเดินทาง การขาย การลงทะเบียน
- แต่ละ entity จะมี attributes เพื่อใช้อธิบายตัวตน เช่น EMPLOYEE มี name, age, address, salary เป็นต้น

- Entity Sets : collection of all entities of a particular entity type

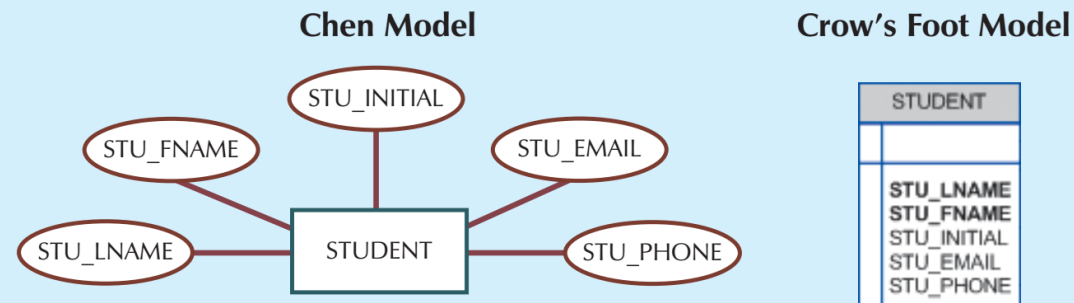


Attributes

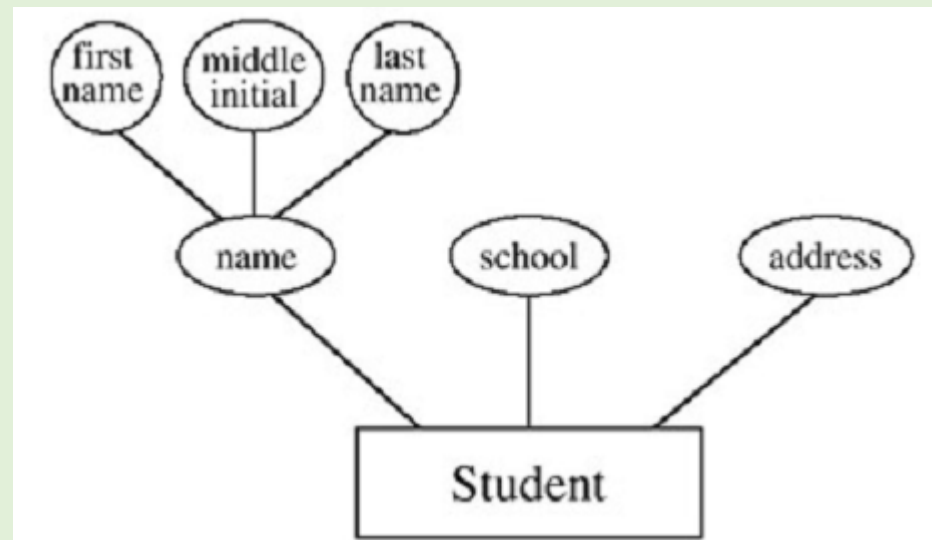
Simple or atomic attributes cannot be further broken down or subdivided, hence the notion "atomic." One can examine the domain of values^[2] of an attribute to elicit whether an attribute is simple or not. An example of a simple or atomic attribute would be Social Security number, where a person would be expected to have only one, undivided Social Security number.

FIGURE 4.1

The attributes of the STUDENT entity: Chen and Crow's Foot



A composite attribute, sometimes called a group attribute, is an attribute formed by combining or aggregating related attributes. The names chosen for composite attributes should be descriptive and general. The concept of



System Database

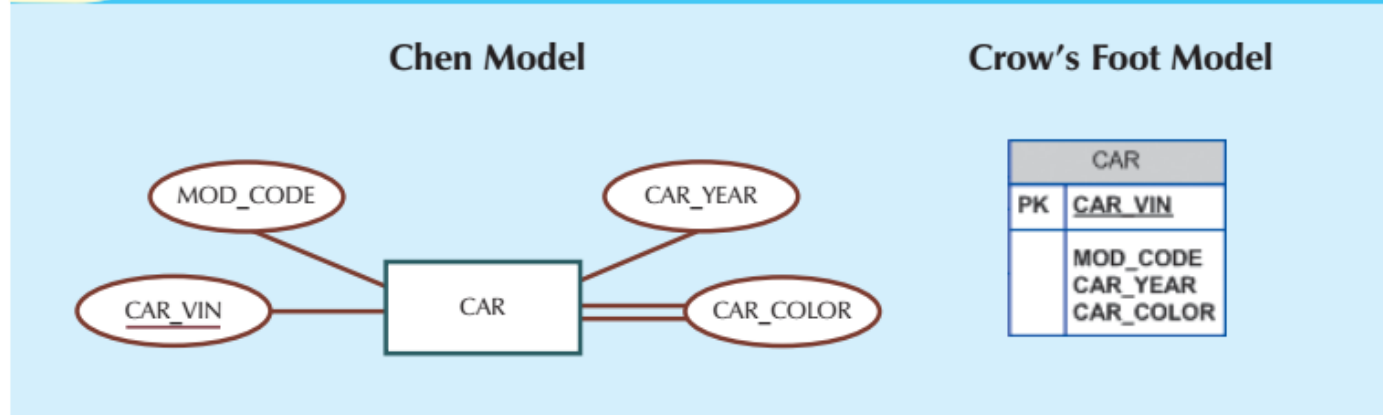
นักตะ

สโเมตร

Another type of non-simple attribute that has to be managed is called a multi-valued attribute. The multi-valued attribute, as the name implies, may take on more than one value for a given occurrence of an entity. For

FIGURE 4.3

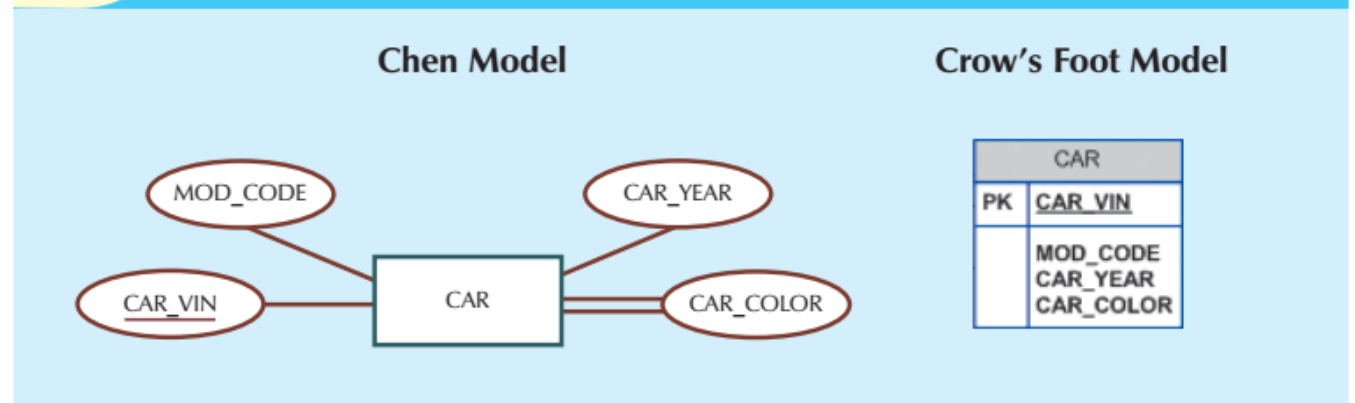
A multivalued attribute in an entity



CAR (CAR_VIN, MOD_CODE, CAR_YEAR, CAR_COLOR)

FIGURE 4.3

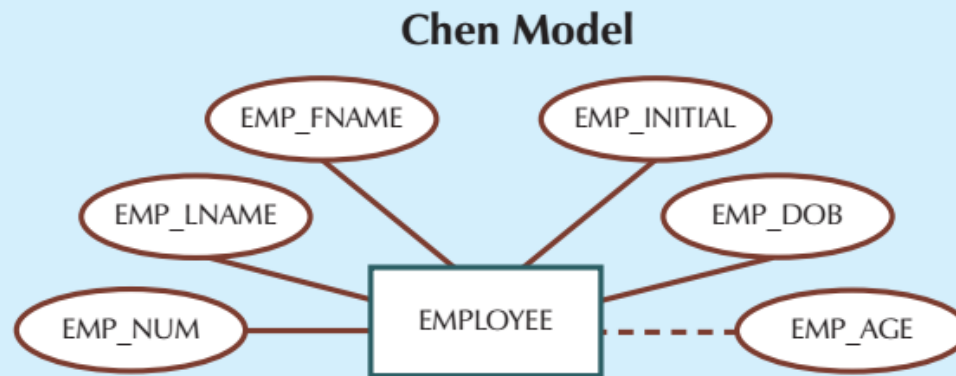
A multivalued attribute in an entity



Derived attributes are attributes that the user may envision but may not be recorded per se. These derived attributes can be calculated from other data in the database. An example of a derived attribute would be an age that could be calculated once a student's birthdate is entered. In the Chen-like

FIGURE 4.6

Depiction of a derived attribute



Crow's Foot Model

EMPLOYEE	
PK	<u>EMP_NUM</u>
	EMP_LNAME
	EMP_FNAME
	EMP_INITIAL
	EMP_DOB
	EMP_AGE

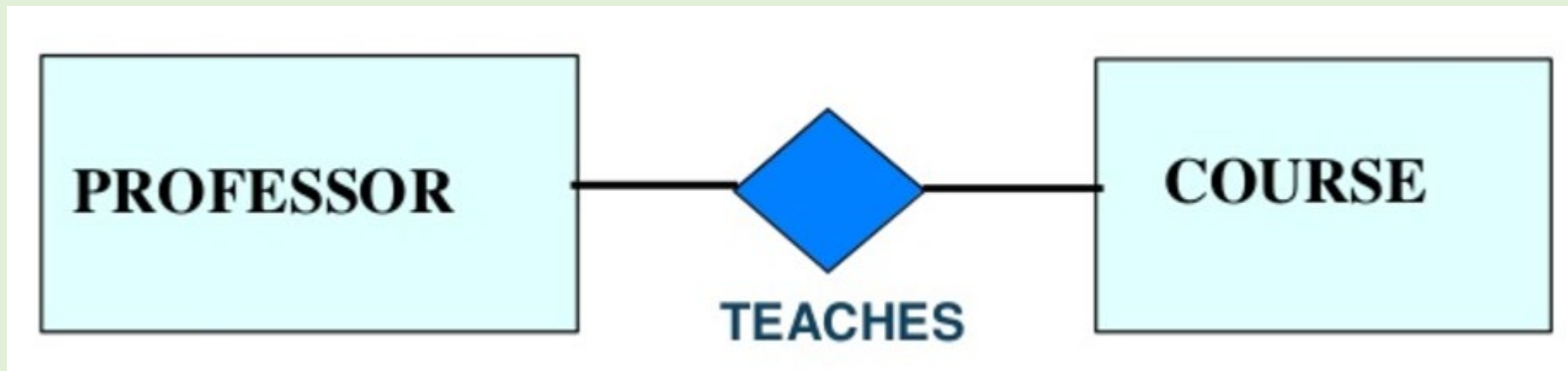
**TABLE
4.2**

Advantages and Disadvantages of Storing Derived Attributes

	DERIVED ATTRIBUTE	
	STORED	NOT STORED
Advantage	<ul style="list-style-type: none">Saves CPU processing cyclesSaves data access timeData value is readily availableCan be used to keep track of historical data	<ul style="list-style-type: none">Saves storage spaceComputation always yields current value
Disadvantage	<ul style="list-style-type: none">Requires constant maintenance to ensure derived value is current, especially if any values used in the calculation change	<ul style="list-style-type: none">Uses CPU processing cyclesIncreases data access timeAdds coding complexity to queries

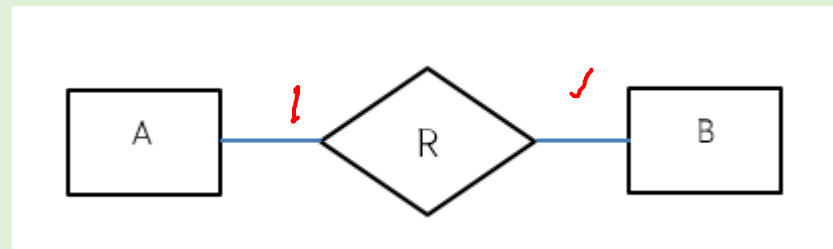
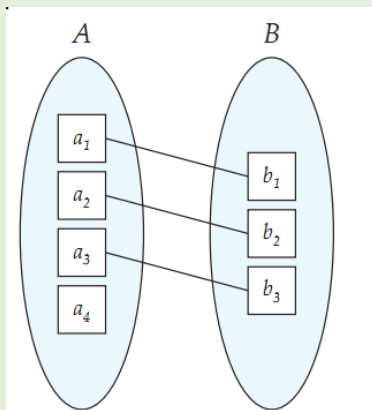
Relationship

- ความสัมพันธ์ของสองเอนทิตีที่เซตเช่นนี้เราเรียกว่าความสัมพันธ์ไบนารี (binary relationship)

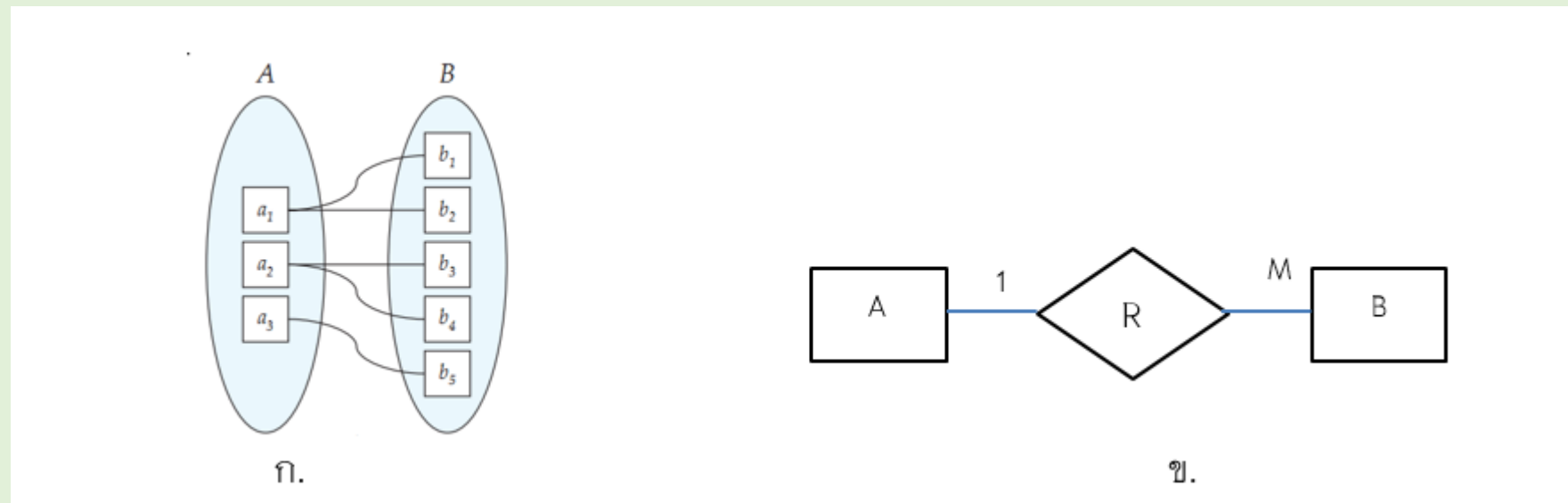


ชนิดของความสัมพันธ์ Binary

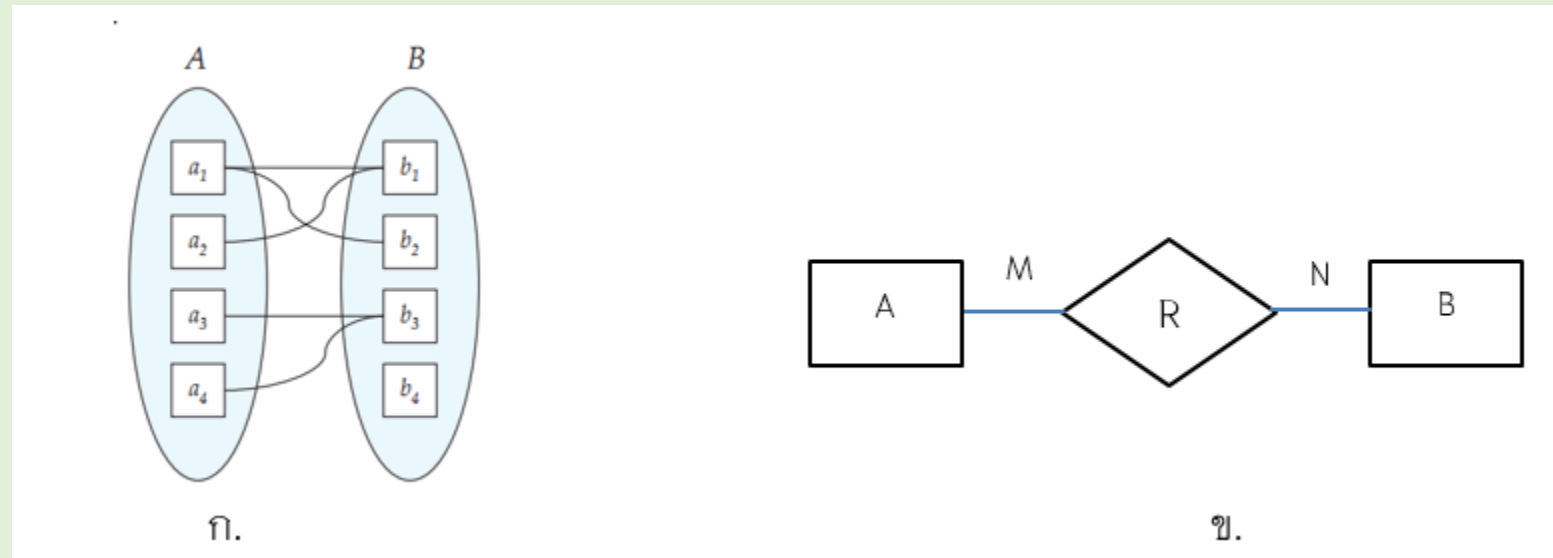
แบบหนึ่งต่อหนึ่ง (1:1) หมายถึง เอนทิตีเซต A มีความสัมพันธ์เพียงหนึ่งความสัมพัธ์กับเอนทิตีเซต B และในทางกลับกันเอนทิตีเซต B ก็มีความสัมพันธ์เพียงหนึ่งความสัมพัธ์กับเอนทิตีเซต A



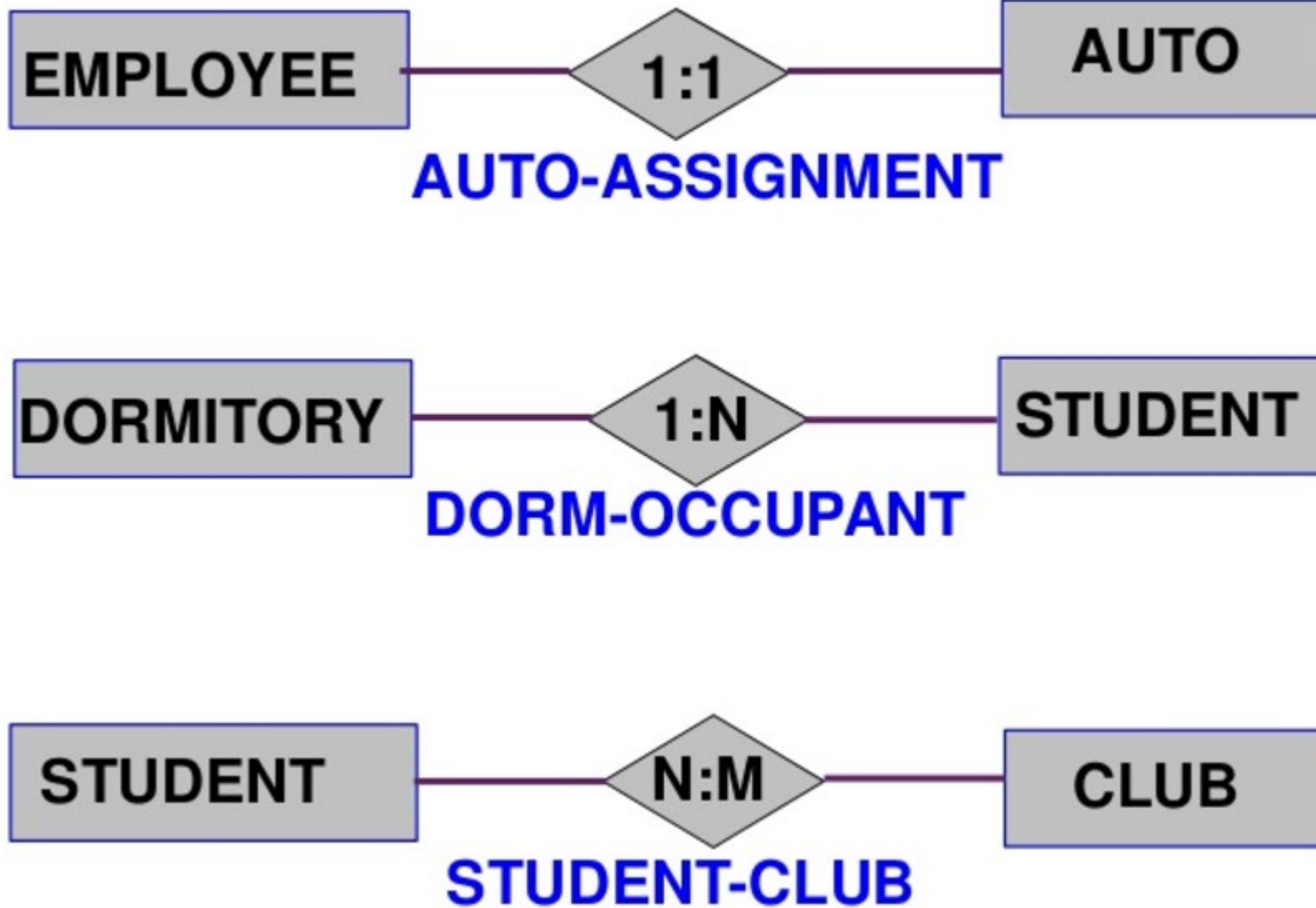
- แบบหนึ่งต่อกลุ่ม (1:M) หมายถึง เอนทิตีเซต A มีความสัมพันธ์แบบไม่จำกัดจำนวนความสัมพันธ์กับเอนทิตี B แต่ในทางกลับกันเอนทิตีเซต B มีความสัมพันธ์เพียงหนึ่งความสัมพันธ์กับเอนทิตีเซต A



- แบบกลุ่มต่อกลุ่ม (M:N) หมายถึง เอนทิตีเซต A มีความสัมพันธ์แบบไม่จำกัดจำนวนความสัมพันธ์กับเอนทิตีเซต B และในทางกลับกันเอนทิตีเซต B มีความสัมพันธ์แบบไม่จำกัดจำนวนความสัมพันธ์กับเอนทิตีเซต A



Clip slide

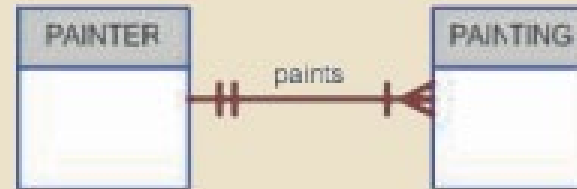


Chen Notation

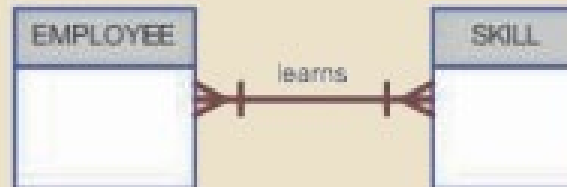
Crow's Foot Notation

UML Class Diagram Notation

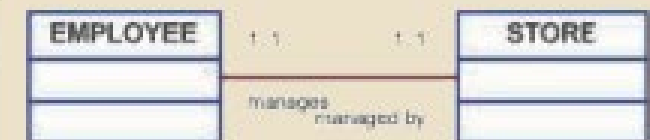
A One-to-Many (1:M) Relationship: a PAINTER can paint many PAINTINGs; each PAINTING is painted by one PAINTER.



A Many-to-Many (M:N) Relationship: an EMPLOYEE can learn many SKILLs; each SKILL can be learned by many EMPLOYEEs.



A One-to-One (1:1) Relationship: an EMPLOYEE manages one STORE; each STORE is managed by one EMPLOYEE.



Business rules

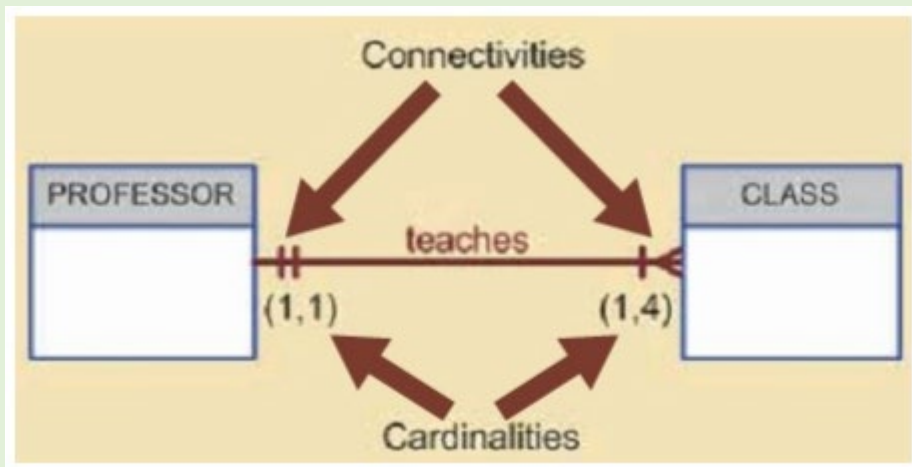
A **business rule** is a brief, precise, and unambiguous description of a policy, procedure, or principle within a specific organization. In a sense, busi-

- A customer may generate many invoices.
- An invoice is generated by only one customer.
- A training session cannot be scheduled for fewer than 10 employees or for more than 30 employees.

ข้อกำหนดเพิ่มเติม

- แบบมีส่วนร่วมทั้งหมด (total participation/mandatory) หมายถึง สมาชิกทุกตัวในเอนทิตีเซตจะต้องมีความสัมพันธ์กับอีกเอนทิตีเซต เช่น เอนทิตีเซต A มีความสัมพันธ์แบบมีส่วนร่วมทั้งหมดกับเอนทิตีเซต B
- แบบมีส่วนร่วมบางส่วน (partial participation/optional) หมายถึง สมาชิกบางตัวในเอนทิตีเซตมีความสัมพันธ์กับอีกเอนทิตีเซต เช่น เอนทิตีเซต B มีความสัมพันธ์แบบมีส่วนร่วมบางส่วนกับเอนทิตีเซต A





Cardinality (x, N) – (min,max)

ของเอ็นทิตีหนึ่งๆจะหมายถึงการที่ไม่มีข้อจำกัดเกี่ยวกับจำนวนสูงสุดของแถวข้อมูลทีี่ซึ่งจะมีความสัมพันธ์กับข้อมูลแถวหนึ่งในอีกตารางหนึ่ง

ตัวอย่าง

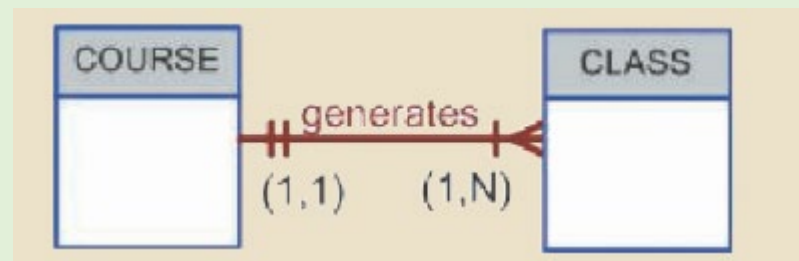
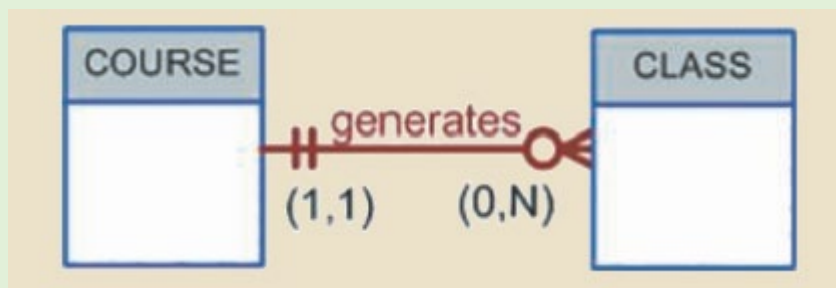
cardinality (1,4) ในฝั่งของเอ็นทิตี **CLASS** จะแสดงว่าอาจารย์คนหนึ่งๆจะต้องสอนอย่างน้อย 1 ชั้นเรียนและสามารถสอนได้มากที่สุด 4 ชั้นเรียน แต่ในส่วนของ

cardinality (1,1) ในฝั่งของ **PROFESSOR** จะแสดงว่าแต่ละชั้นเรียนจะถูกสอนโดยอาจารย์เพียงคนเดียวเท่านั้น

Database System

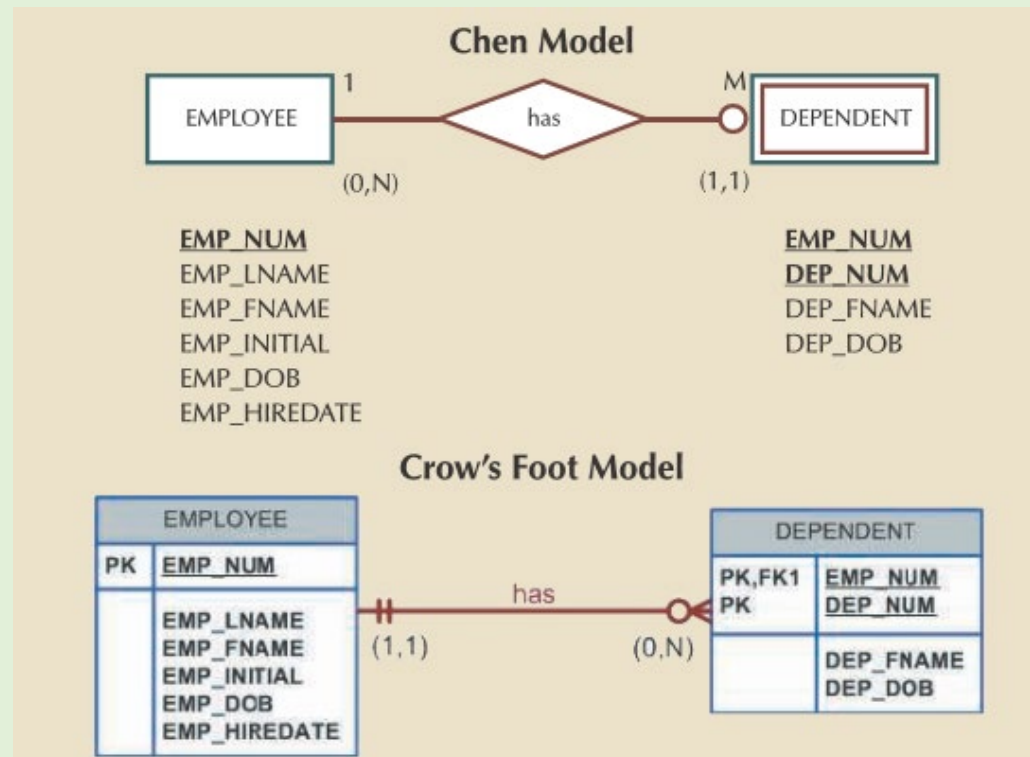


Prof 1 คน ไม่ต้องสอนหรือสอนสูงสุด 3 วิชา
Class 1 วิชา มี prof สอนได้สูงสุด 1 คน



Weak entities

1. The entity is existence-dependent; it cannot exist without the entity with which it has a relationship.
2. The entity has a primary key that is partially or totally derived from the parent entity in the relationship.



Database System

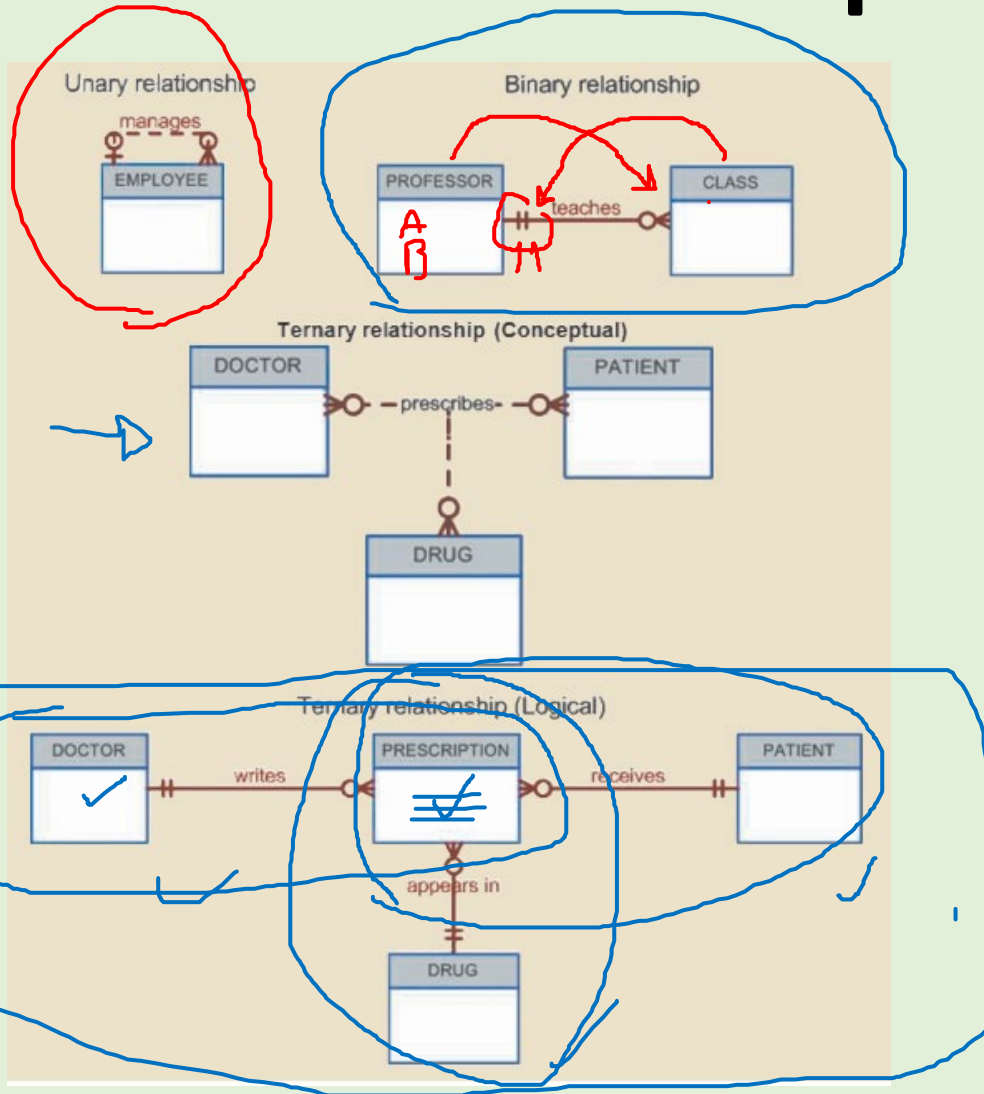
Table name: EMPLOYEE

Database name: Ch04_ShortCo

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_DOB	EMP_HIREDATE
1001	Callifante	Jeanine	J	12-Mar-64	25-May-97
1002	Smithson	William	K	23-Nov-70	28-May-97
1003	Washington	Herman	H	15-Aug-68	28-May-97
1004	Chen	Lydia	B	23-Mar-74	15-Oct-98
1005	Johnson	Melanie		28-Sep-66	20-Dec-98
1006	Ortega	Jorge	G	12-Jul-79	05-Jan-02
1007	O'Donnell	Peter	D	10-Jun-71	23-Jun-02
1008	Brzenski	Barbara	A	12-Feb-70	01-Nov-03

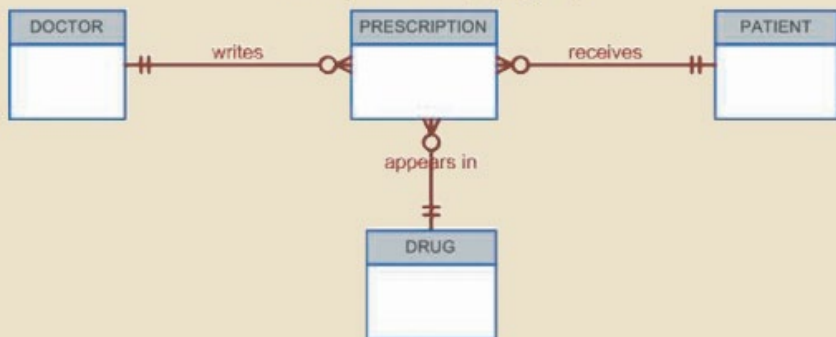
Table name: DEPENDENT

EMP_NUM	DEP_NUM	DEP_FNAME	DEP_DOB
1001		Annelise	05-Dec-97
1001	2	Jorge	30-Sep-02
1003		Suzanne	25-Jan-04
1006	1	Carlos	25-May-01
1008	1	Michael	19-Feb-95
1008	2	George	27-Jun-98
1008	3	Katherine	18-Aug-03



Database System

Ternary relationship (Logical)



Database name: Ch04_Clinic

Table name: DRUG

DRUG_CODE	DRUG_NAME	DRUG_PRICE
AF15	Atgapan-15	25.00
AF25	Atgapan-25	35.00
DRO	Drosalene Chloride	111.89
DRZ	Druzocholar Cryptolene	18.99
KO15	Kolabar Oxyhexalene	65.75
OLE	Oleander-Dritzapan	123.95
TRYP	Tryptolac Heptadimetric	79.45

Table name: PATIENT

PAT_NUM	PAT_TITLE	PAT_LNAME	PAT_FNAME	PAT_INITIAL	PAT_DOB	PAT_AREACODE	PAT_PHONE
100	Mr.	Kolmycz	George	D	15-Jun-1942	615	324-5456
101	Ms.	Lewis	Rhonda	G	19-Mar-2005	615	324-4472
102	Mr.	Vandam	Rhett		14-Nov-1958	901	675-8993
103	Ms.	Jones	Anne	M	16-Oct-1974	615	898-3456
104	Mr.	Lange	John	P	08-Nov-1971	901	504-4430
105	Mr.	Williams	Robert	D	14-Mar-1975	615	890-3220
106	Mrs.	Smith	Jeanine	K	12-Feb-2003	615	324-7883
107	Mr.	Diante	Jorge	D	21-Aug-1974	615	890-4567
108	Mr.	Wlesenbach	Paul	R	14-Feb-1966	615	897-4358
109	Mr.	Smith	George	K	18-Jun-1961	901	504-3339
110	Mrs.	Genkazi	Leighla	vV	19-May-1970	901	569-0093
111	Mr.	Washington	Rupert	E	03-Jan-1966	615	890-4925
112	Mr.	Johnson	Edward	E	14-May-1961	615	898-4387
113	Ms.	Smythe	Melanie	P	15-Sep-1970	615	324-9006
114	Ms.	Brandon	Marie	G	02-Nov-1932	901	882-0845
115	Mrs.	Seranda	Hermine	R	25-Jul-1972	615	324-5505
116	Mr.	Smith	George	A	08-Nov-1965	615	890-2984

Table name: DOCTOR

DOC_ID	DOC_LNAME	DOC_FNAME	DOC_INITIAL	DOC_SPECIALTY
29827	Sanchez	Julio	J	Dermatology
32445	Jorgensen	Annelise	G	Neurology
33456	Korenski	Anstoly	A	Urology
33989	LeGrande	George		Pediatrics
34409	Washington	Dennis	F	Orthopaedics
36221	McPherson	Katye	H	Dermatology
36712	Dreifag	Herman	G	Psychiatry
38995	Minh	Tran		Neurology
40004	Chin	Ming	D	Orthopaedics
40028	Feinstein	Denise	L	Gynecology

Table name: PRESCRIPTION

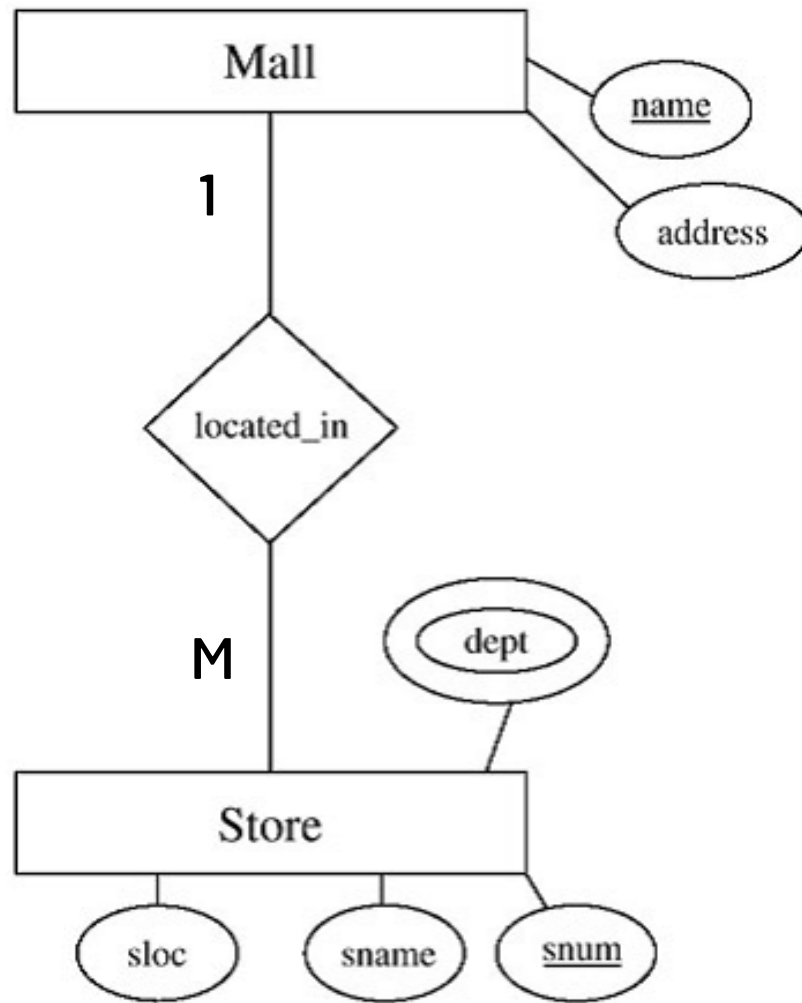
DOC_ID	PAT_NUM	DRUG_CODE	PRES_DOSAGE	PRES_DATE
32445	102	DRZ	2 tablets every four hours -- 50 tablets total	12-Nov-16
32445	113	OLE	1 teaspoon with each meal -- 250 ml total	14-Nov-16
34409	101	KO15	1 tablet every six hours -- 30 tablets total	14-Nov-16
36221	109	DRO	2 tablets with every meal -- 60 tablets total	14-Nov-16
38995	107	KO15	1 tablet every six hours -- 30 tablets total	14-Nov-16

Case Study

Case Study: West Florida Mall

A new mall, West Florida Mall, just had its grand opening three months ago in Pensacola, Florida. This new mall is attracting a lot of customers and stores. West Florida Mall, which is part of a series of malls owned by a parent company, now needs a database to keep track of the management of the mall in terms of keeping track of all its stores as well as the owners and workers of the stores. Before we build a database for this system of malls, the first step will be to design an ER diagram for the mall owner. We gathered the following initial user specifications about the malls, with which we can start creating our the ER diagram:

- We need to record information about the mall and each store in the mall. We will need to record the mall's name and address. A mall, at any point in time, must contain one or more stores.
- For each store, we will need to keep the following information: store number (which will be unique), the name of the store, the location of the store (room number), departments, the owner of the store, and manager of the store. Each store may have more than one department, and each department is managed by a manager. Each store will have only one store manager. Each store is owned by only one owner. Each store is located in one and only one mall.
- A store manager can manage only one store. We have to record information on the store manager: the name, social security number, which store he or she is working for, and salary.
- The store owner is a person. We have to record information about the store owner, such as name, social security number, address, and office phone number. A store owner has to own at least one store, and may own more than one store.



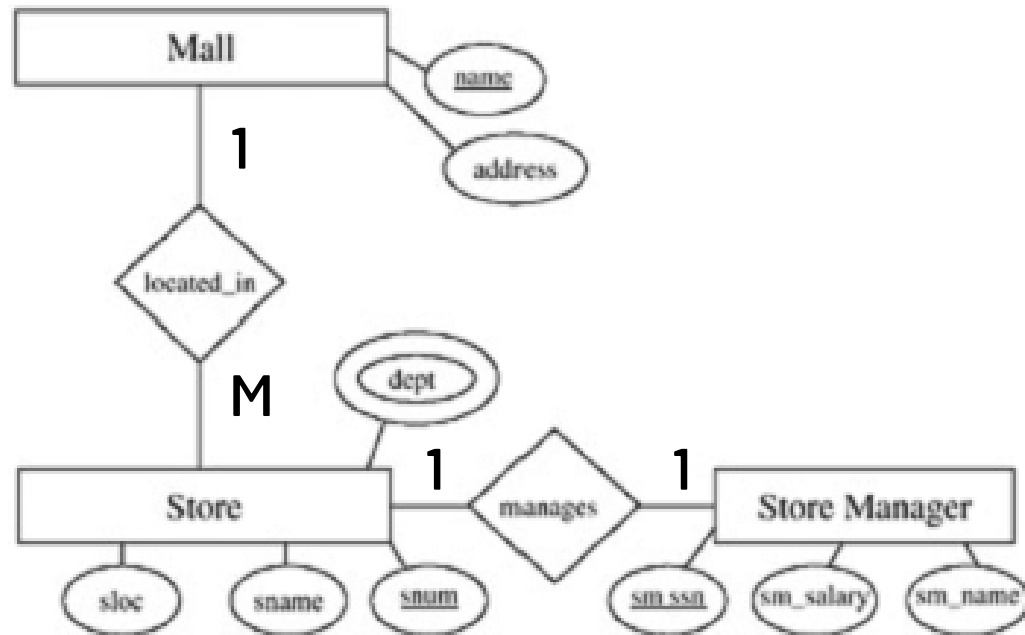
The Attributes for STORE

For each STORE, there will always be one and only one sname (store name). The value for sname will not be subdivided.

For each STORE, there will always be one and only one snum (store number). The value for snum will be unique, and not be subdivided.

For each STORE, we will record a sloc (store location). There will be one sloc recorded for each STORE. The value for sloc will not be subdivided.

For each STORE, we will record depts (departments). There will be more than one depts recorded for each STORE. The value for depts will not be subdivided.

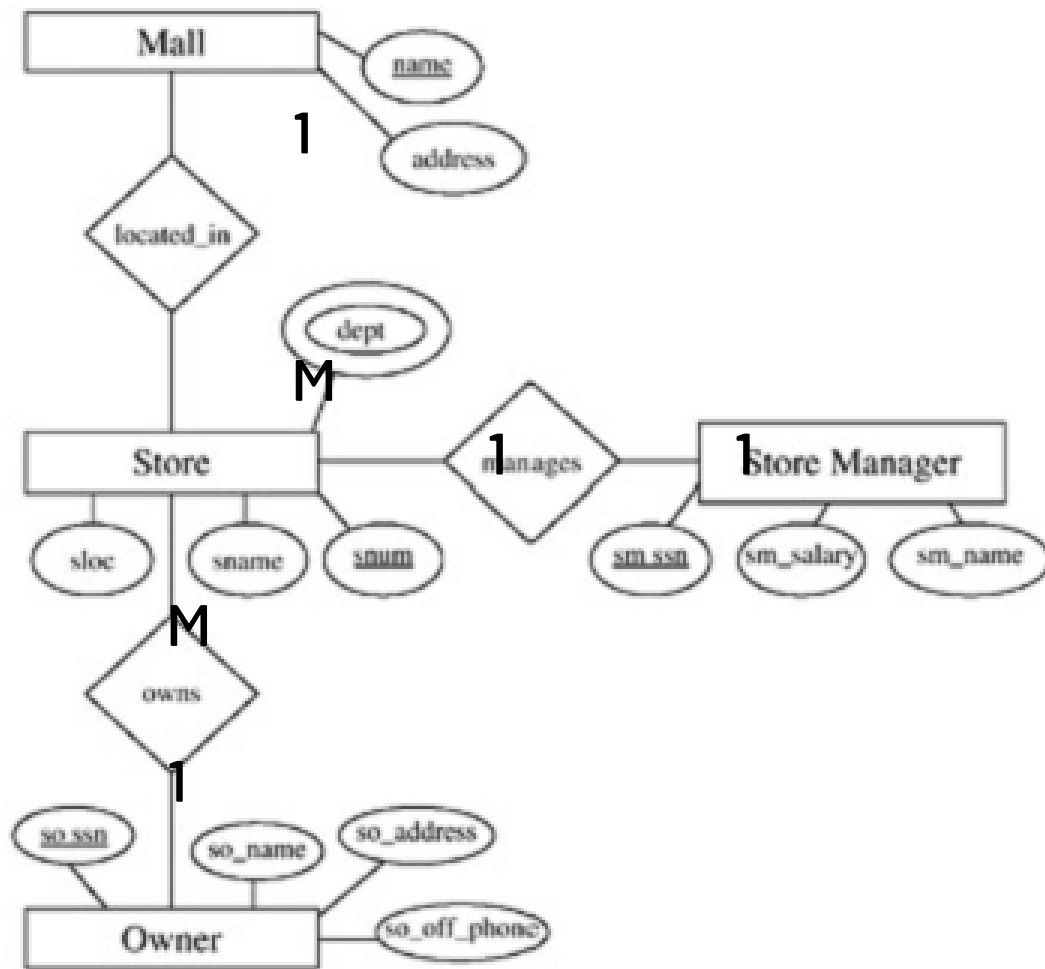


The Attributes for STORE_MANAGER

For each STORE_MANAGER, there will always be one and only one sm_name (store manager name). The value for sm_name will not be subdivided.

For each STORE_MANAGER, there will always be one and only one sm_ssn (store manager ssn). The value for sm_ssn will be unique, and not be subdivided.

For each STORE_MANAGER, we will record a sm_salary (store manager salary). There will be one and only one sm_salary recorded for each STORE_MANAGER. The value for sm_salary will not be subdivided.



The Attributes for STORE_OWNER

For each STORE_OWNER, there will always be one and only one so_name (store owner name). The value for so_name will not be subdivided.

For each STORE_OWNER, there will always be one and only one so_ssn (store owner ssn). The value for so_ssn will be unique, and will not be subdivided.

For each STORE_OWNER, there will always be one and only one so_off_phone (store owner office phone). The value for so_off_phone will be unique, and will not be subdivided.

For each STORE_OWNER, we will record a so_address (store owner address). There will be one and only one so_address recorded for each STORE_OWNER. The value for so_address will not be subdivided.

การบ้านให้ทุกคนเขียน
West Florida mall โดยใช้ Craw's foot

Sample Database

- The company is organized into departments. Each department has a unique name, a unique number, and a particular employee who manages the department. We keep track of the start date when that employee began managing the department. A department may have several locations.
- A department controls a number of projects, each of which has a unique name, a unique number, and a single location.
- The database will store each employee's name, Social Security number,² address, salary, sex (gender), and birth date. An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department. It is required to keep track of the current number of hours per week that an employee works on each project, as well as the direct supervisor of each employee (who is another employee).
- The database will keep track of the dependents of each employee for insurance purposes, including each dependent's first name, sex, birth date, and relationship to the employee.

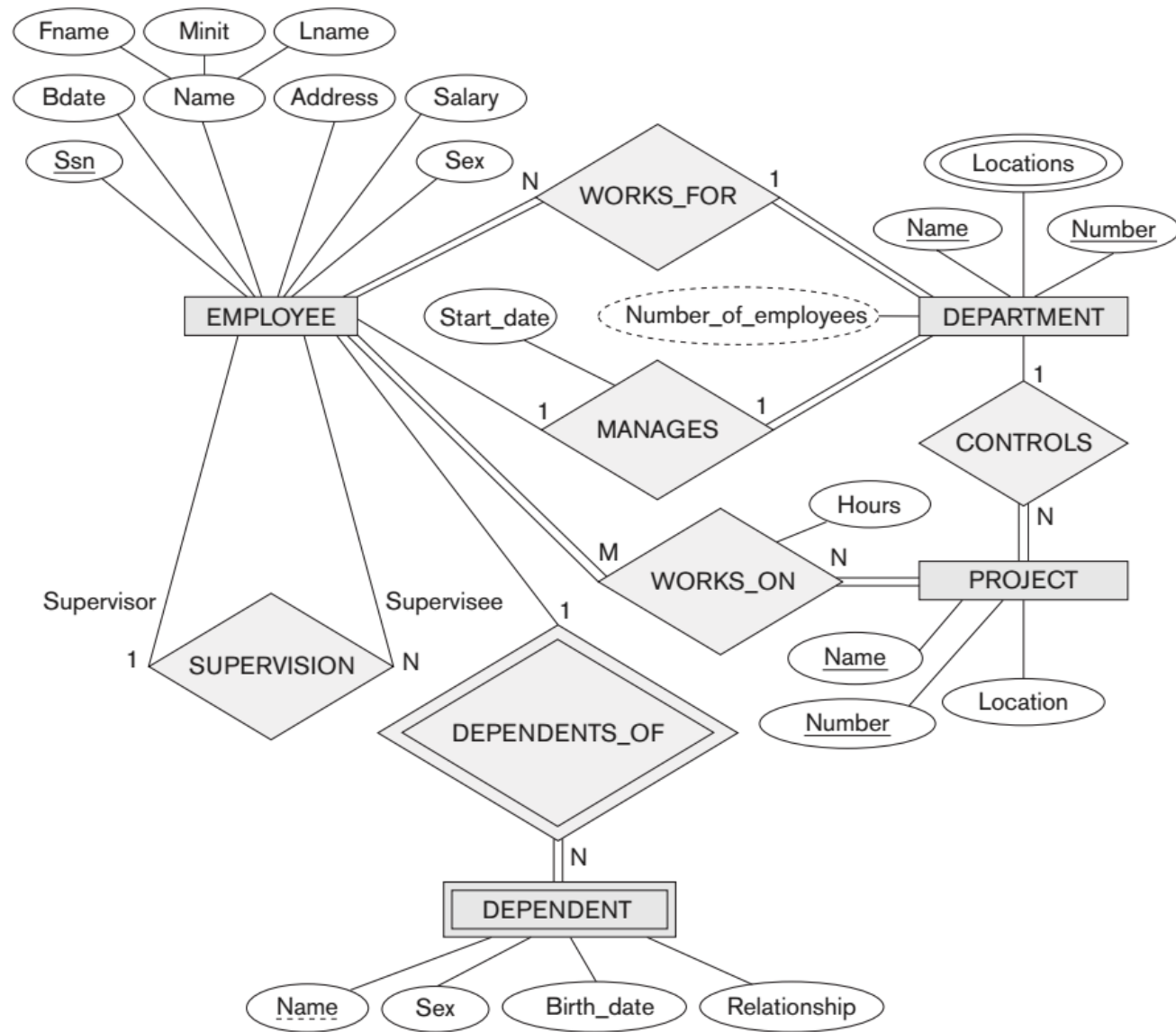
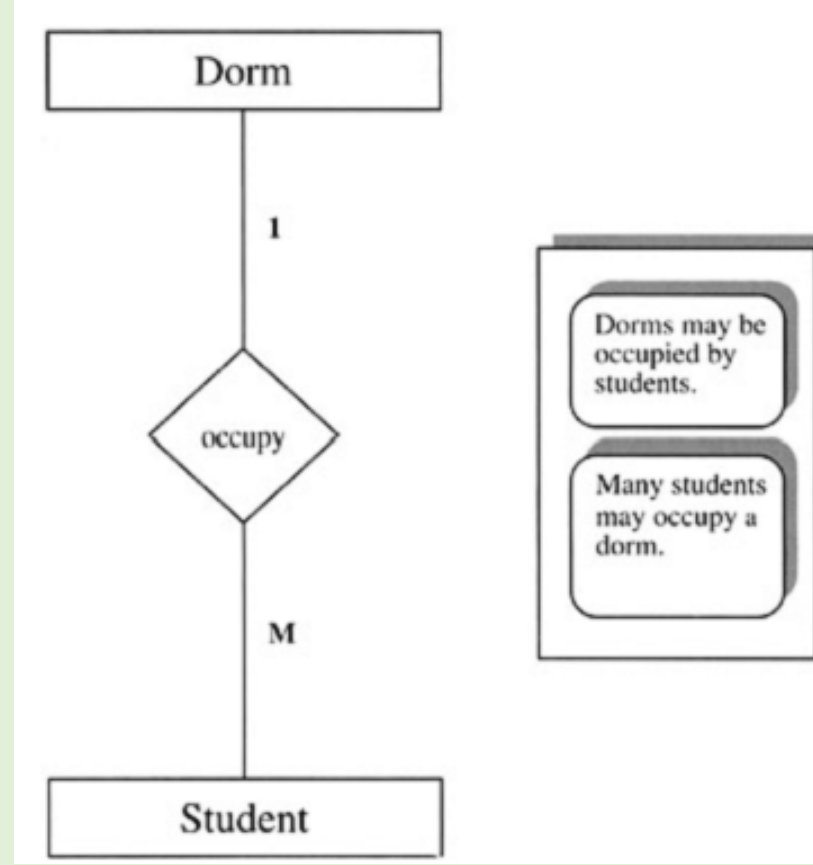


Figure 3.2
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter and is summarized in Figure 3.14.

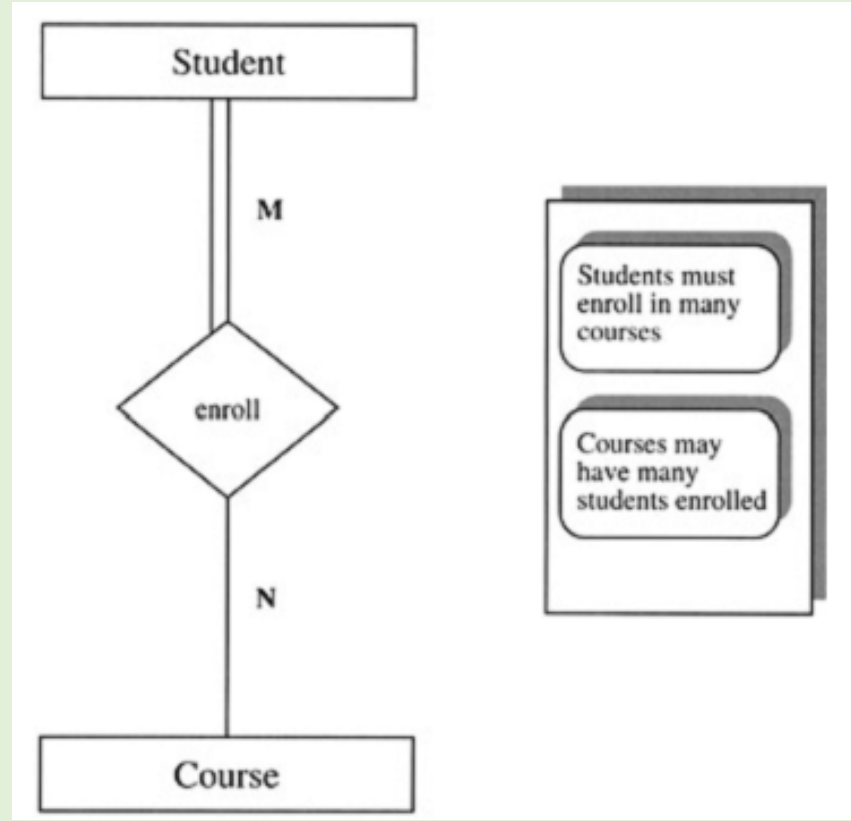
Student and Dormitory

System Database



Student and Course

System Database

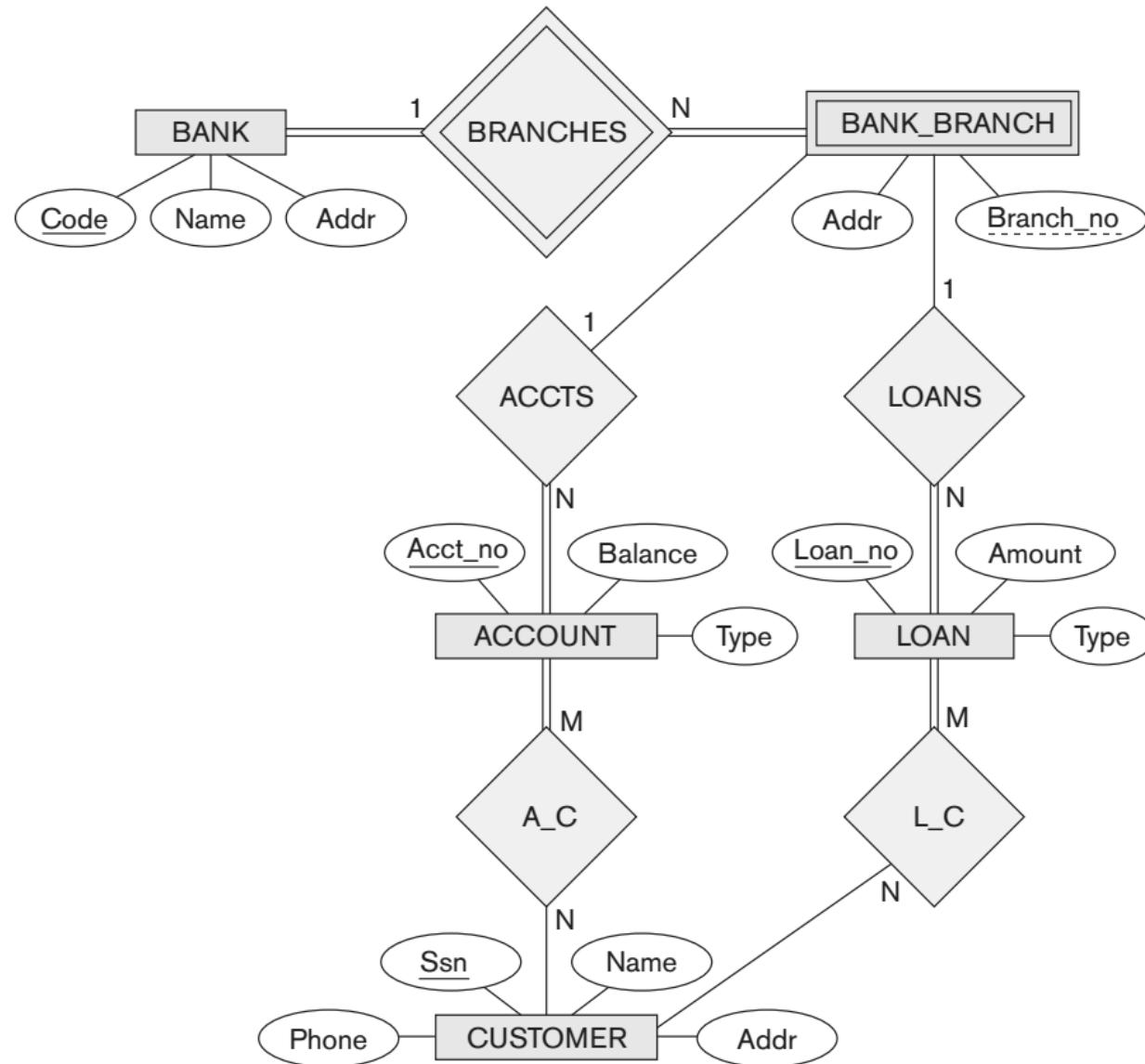


System Database

Banking

Figure 3.22

An ER diagram for a BANK database schema.



System Database

Movies

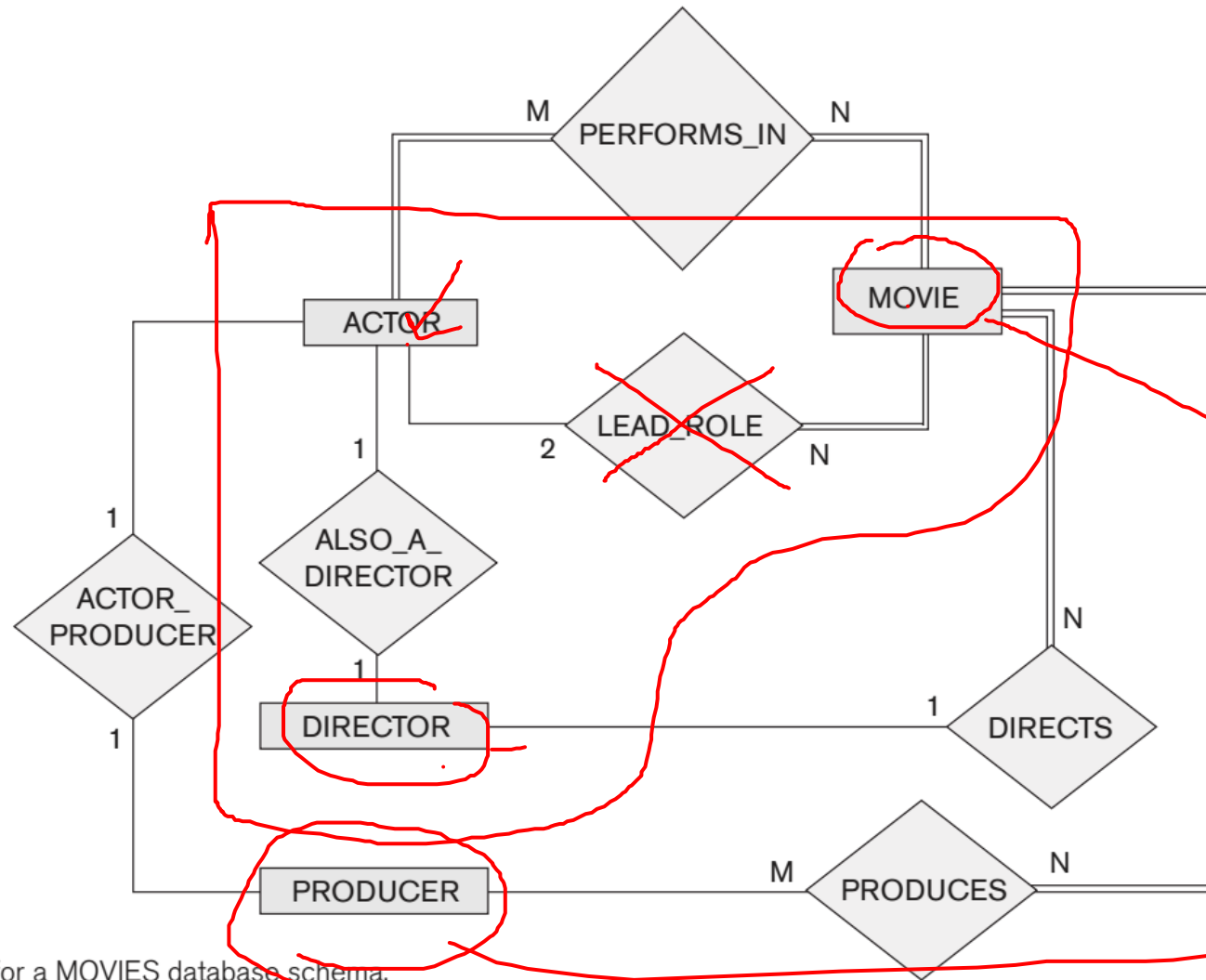


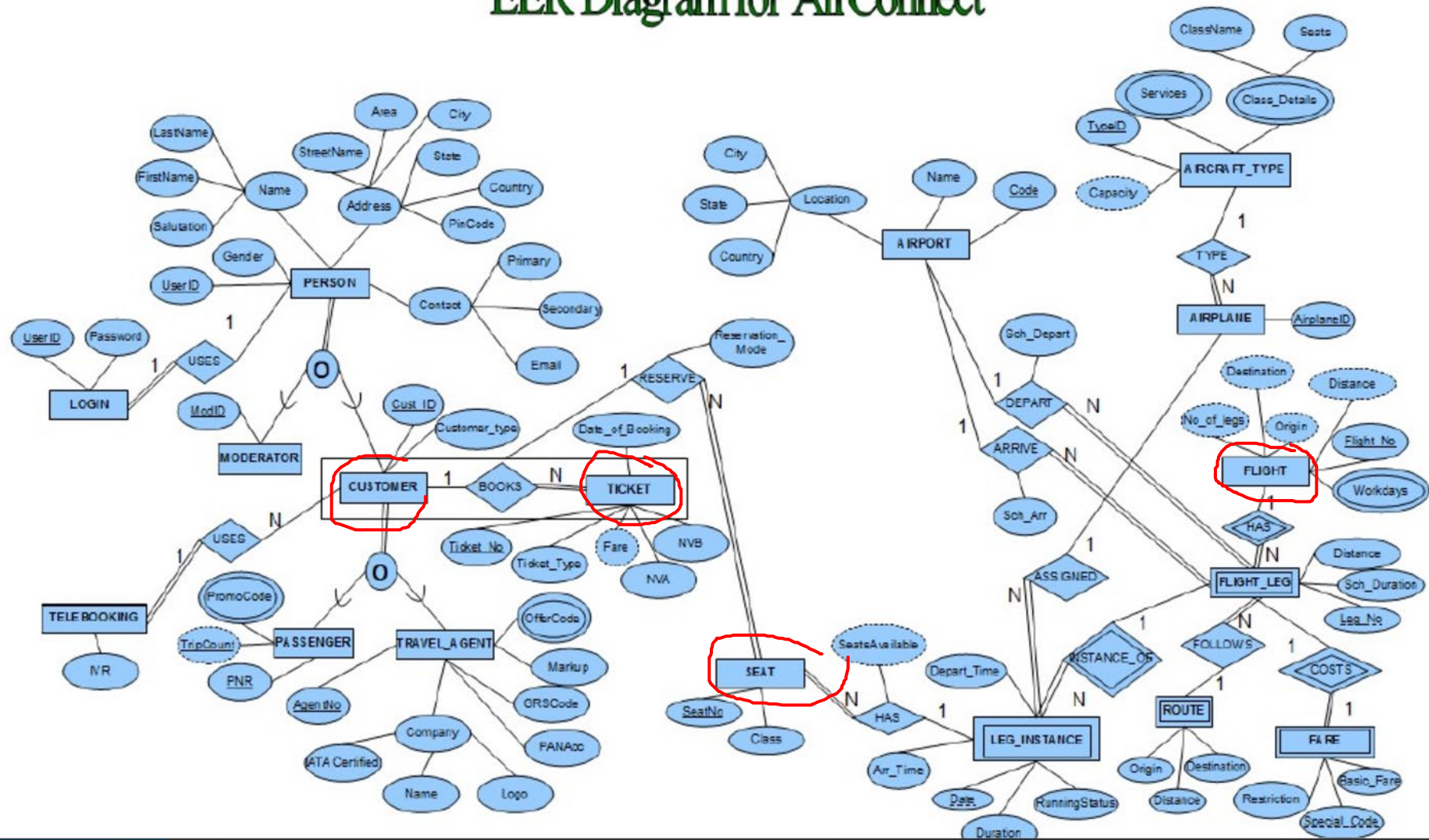
Figure 3.25

An ER diagram for a MOVIES database schema.

Airline

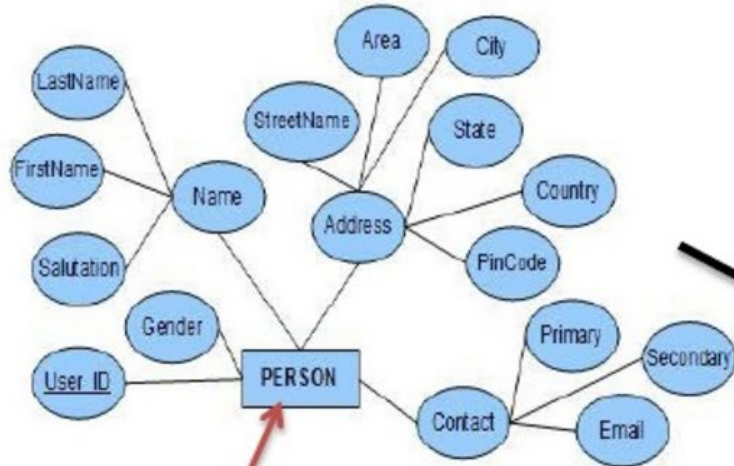
Conceptual Schema

EER Diagram for AirConnect



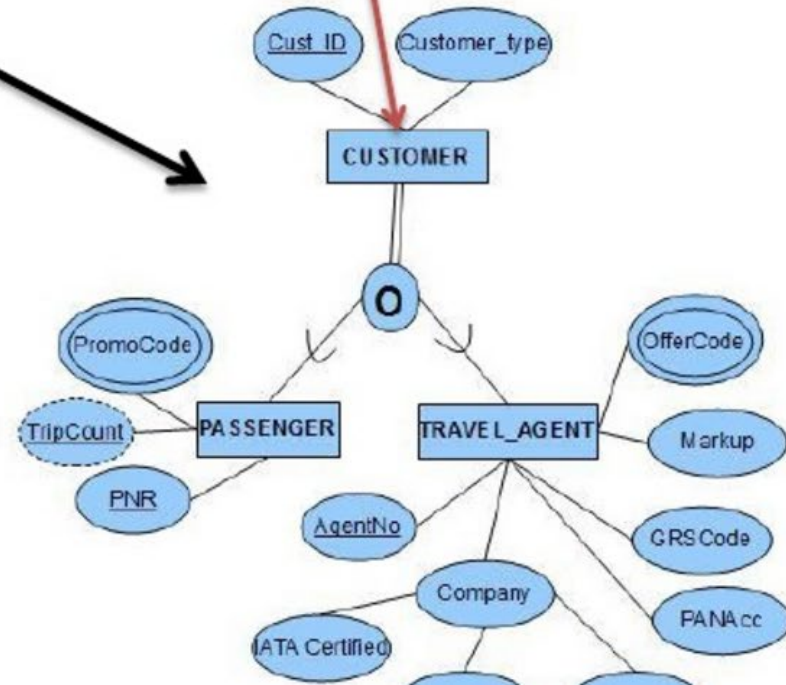
Mandatory Features

All details related to Customers - **PASSENGER, TRAVEL_AGENT**

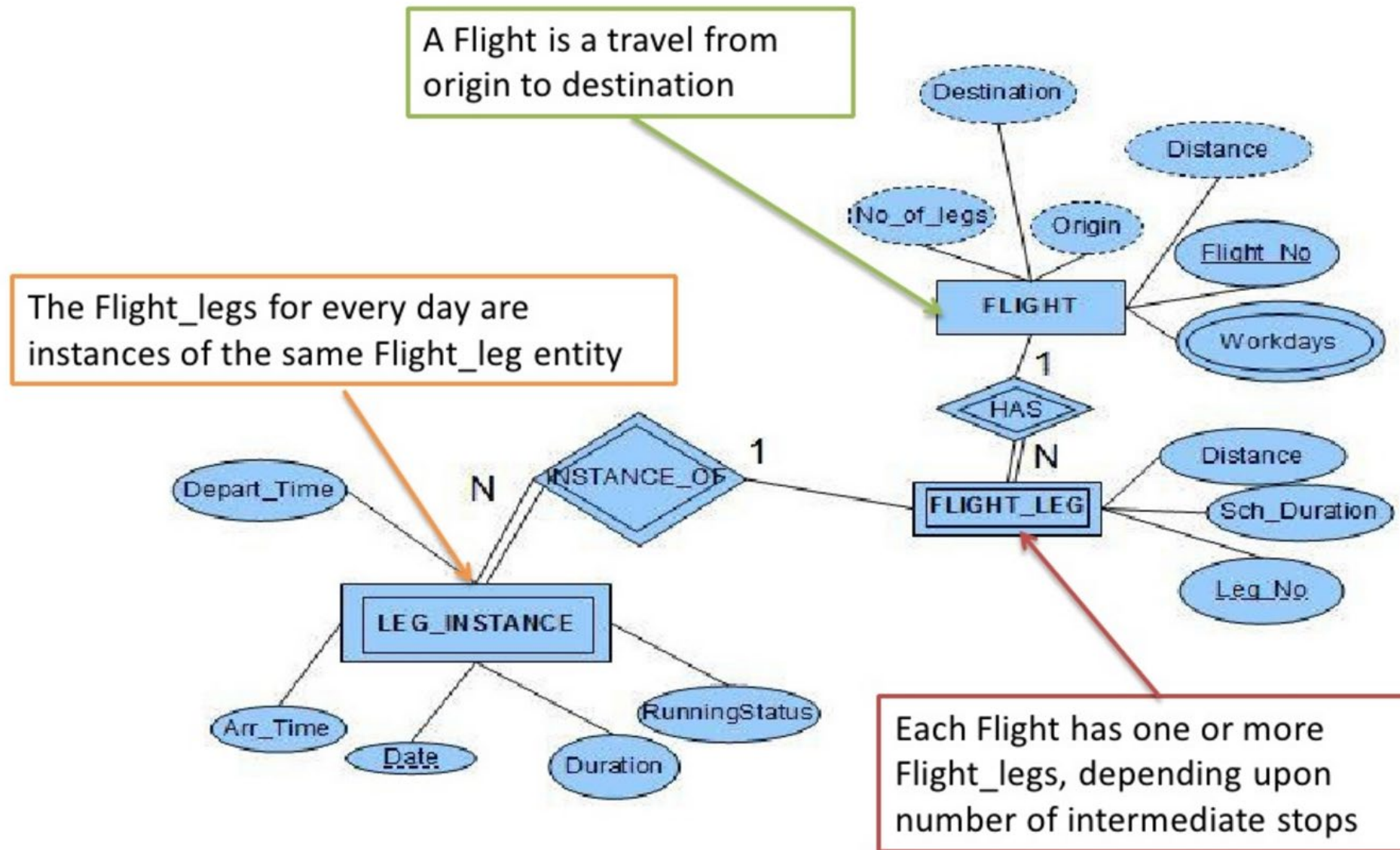


Person – Base Entity : Has all the details like Name, Address, etc.

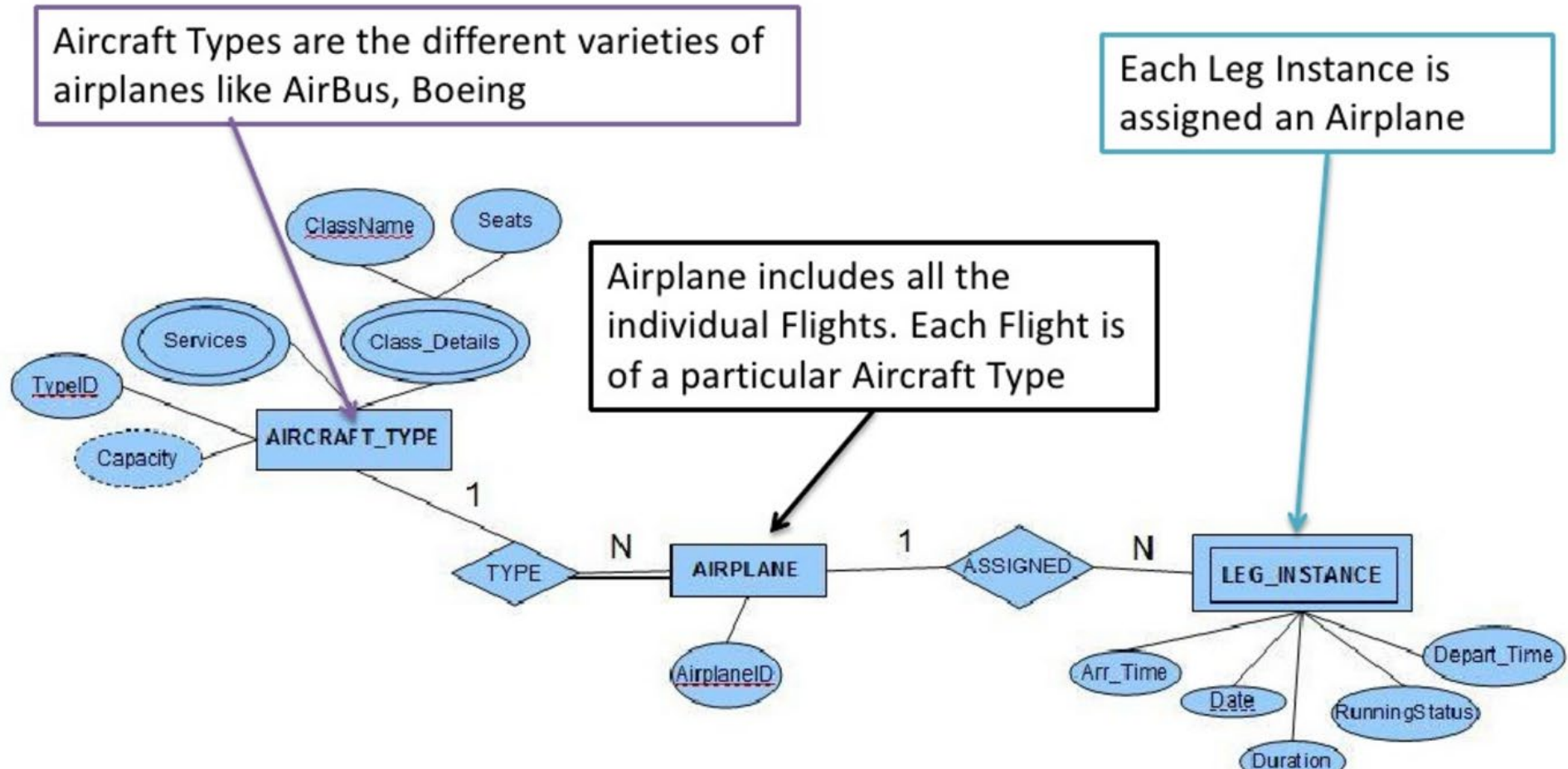
Customer Details – Derived :
Further specializes to Passenger
and Travel Agent



All details related to Flight - **FLIGHT, FLIGHT_LEG, LEG_INSTANCE**



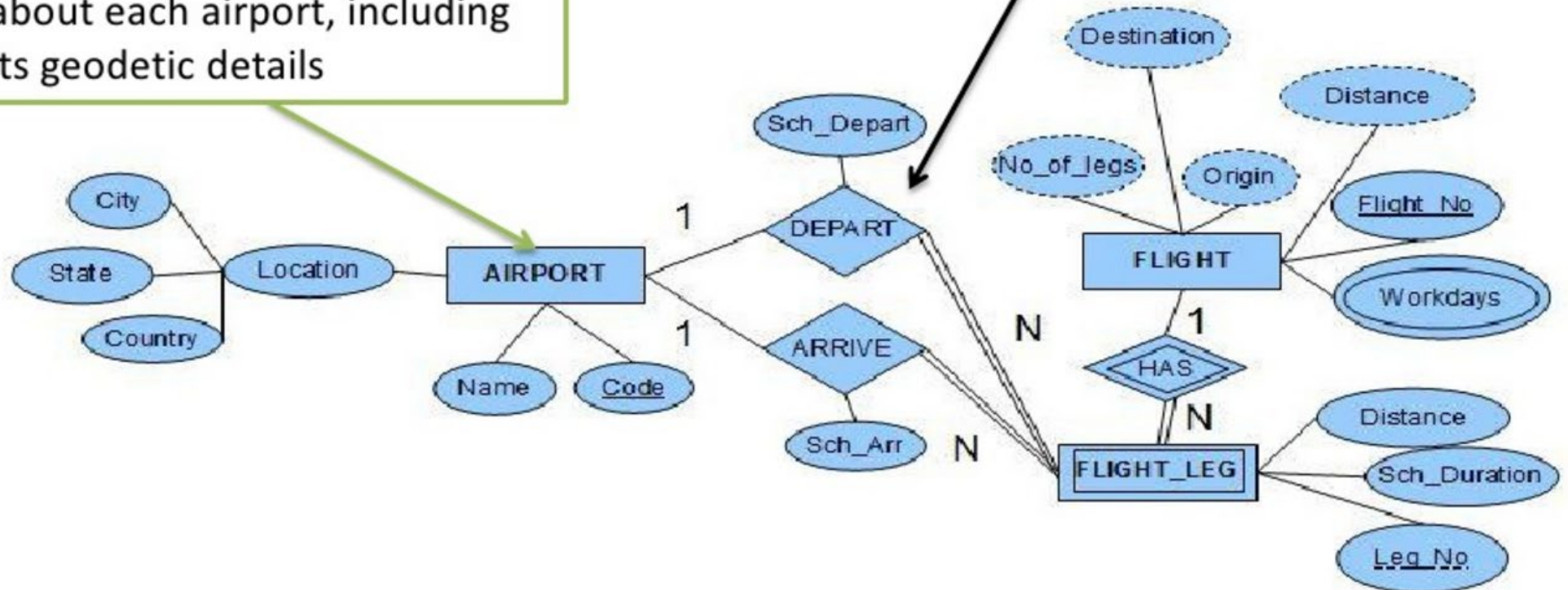
- Details about Airplane : AIRCRAFT_TYPE, AIRPLANE, LEG_INSTANCE



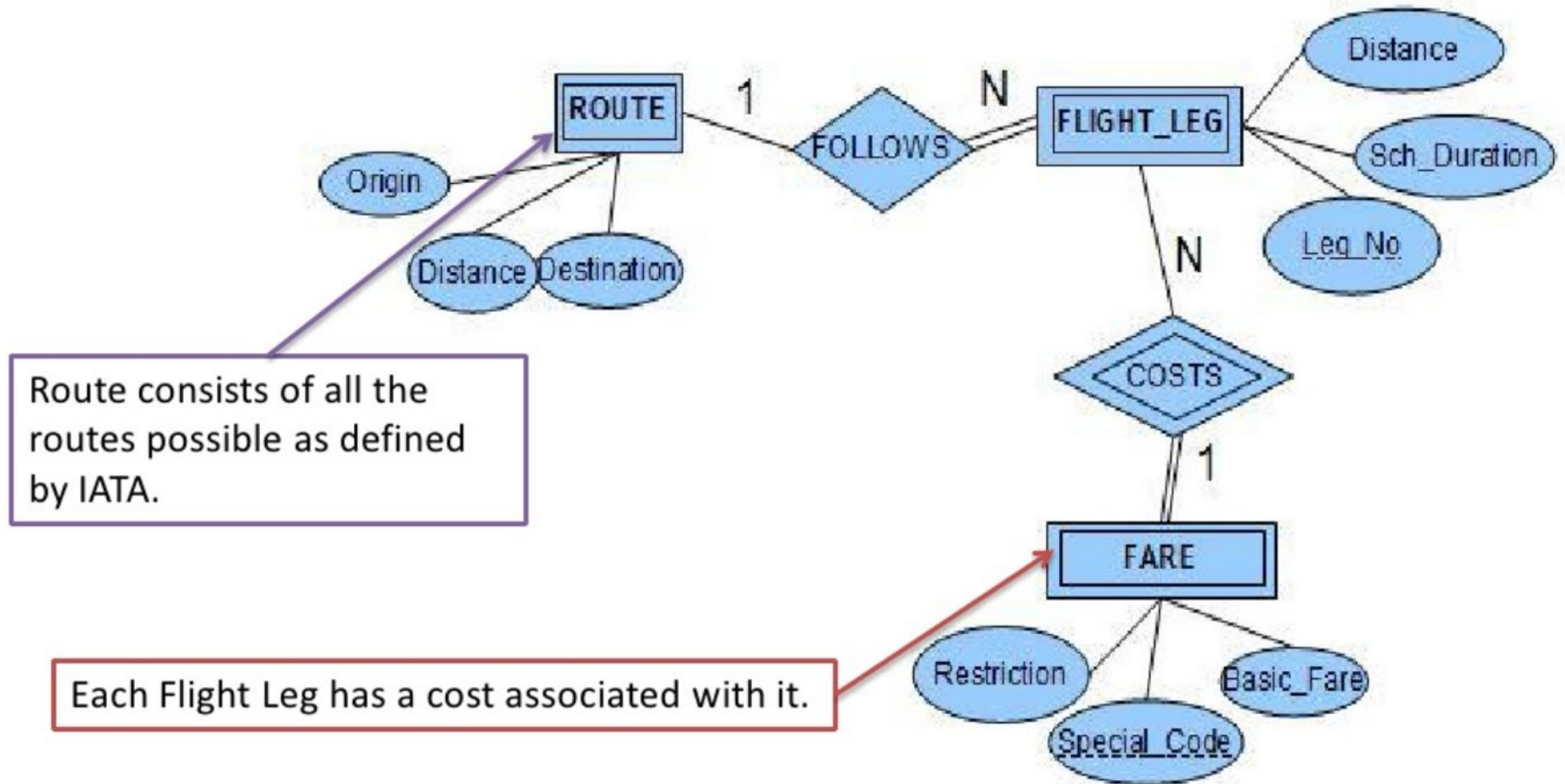
Airports – AIRPORT

Airport entity holds the details about each airport, including its geodetic details

Each Flight Leg must have an airport as the origin and destination.



- Origins and Destination – **ROUTE**



Relational Schema

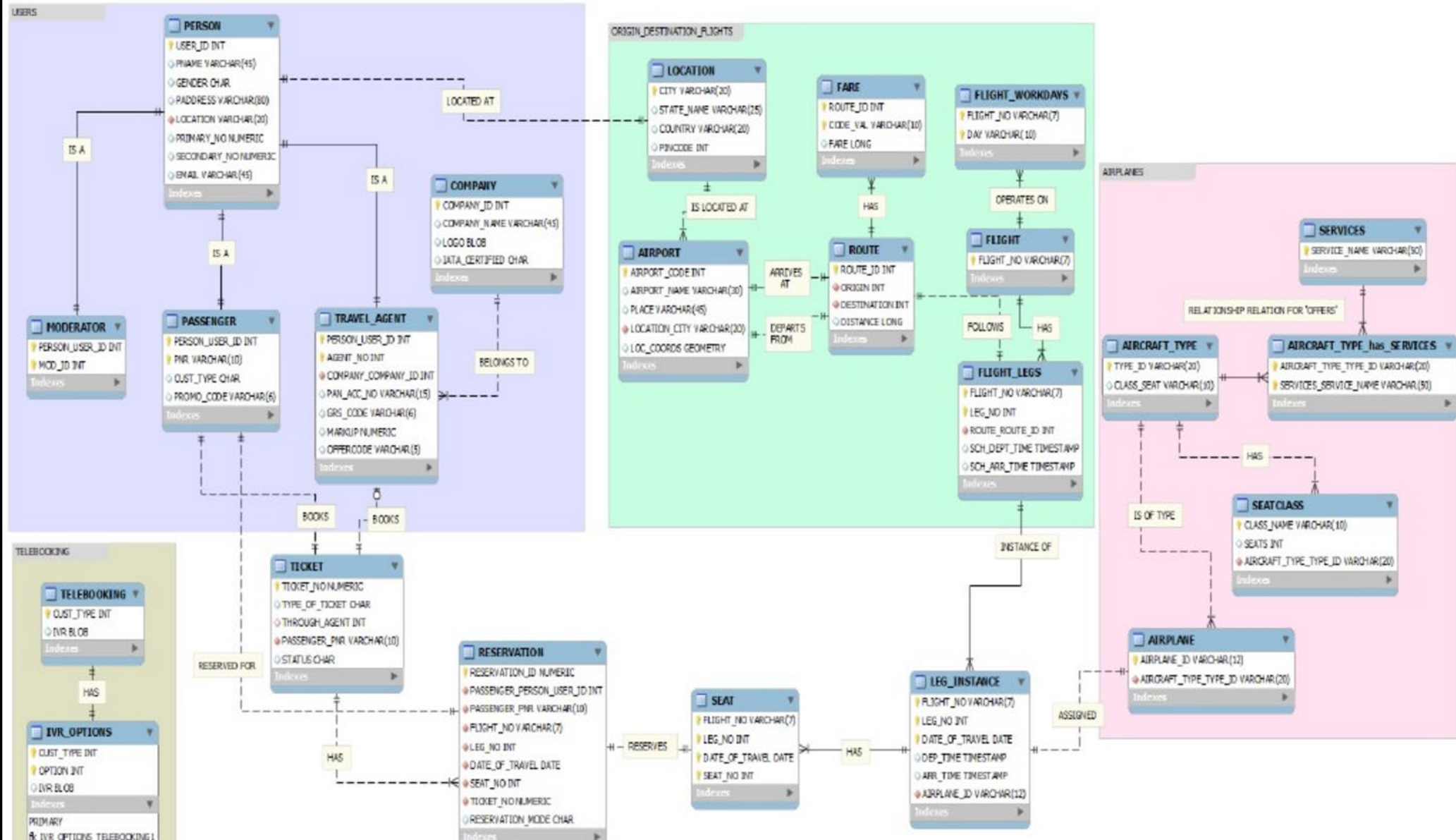


FIGURE 4.35 THE COMPLETED TINY COLLEGE ERD

