DATABASE

Öğr. Gör. Evgin GÖÇERİ

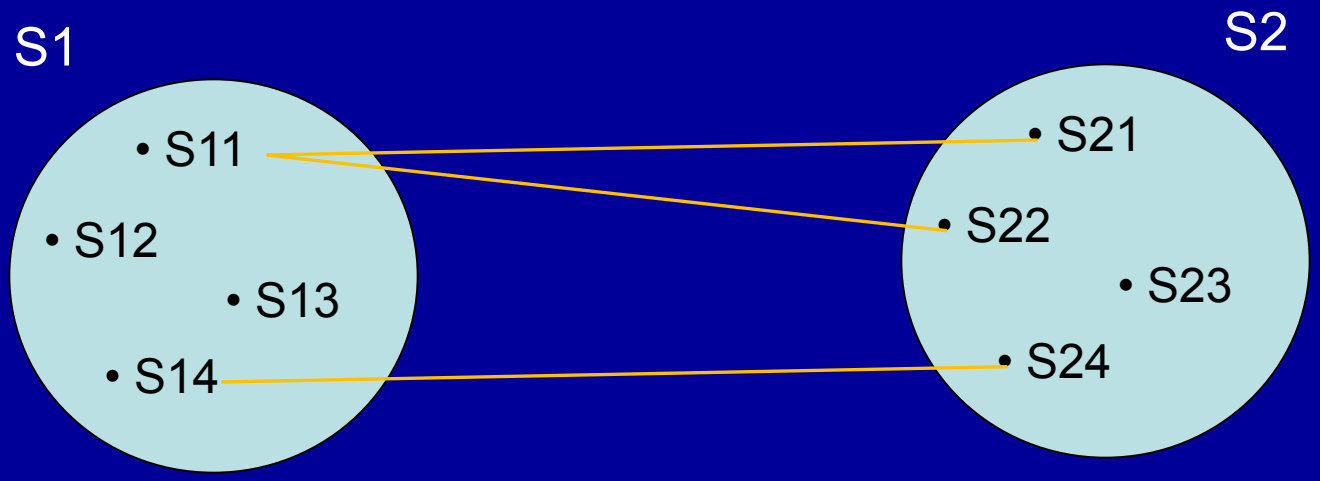
22 Ekim 2014

CENG 301
Normalization & E/R Diagrams

Functional Mapping-1

- A functional mapping or function can be defined as $f:D \rightarrow R$ where D is domain and R is range
- $f:D \rightarrow R$ means that for each value in D, there corresponds a unique value in R
- If every value in D maps to R, then *f* is called as *total function*, otherwise *f* is *partial function*

Functional Mapping-2



- S1 → S2 is Nonfunctional
- S2 → S1 is Partially Functional

Functional Mapping-3

S1		S2	Mapping
One	→	Many	Nonfunctional
One	↔	One	Functional in either direction
Many	→	One	Functional
Many	↔	Many	Nonfunctional

Dependency Theory

- A functional mapping determines uniqueness on the range side

- The role of the relational design and dependency theory is to identify semantics

Functional Dependency -1

- Example : SSN $\longrightarrow$ SALARY

  implies that for each person has only one salary in the company but several people may earn the same amount

- In SSN $\longrightarrow$ SALARY; SSN determines SALARY or
  SALARY depends on SSN

Functional Dependency -2

- A functional dependency is a relationship between or among attributes. Suppose that if we are given the value of one attribute, we can obtain the value of another attribute

- In more general terms, attribute Y is functionally dependent on attribute X, if the value of X determines the value of Y

Functional Dependency -3

- Test

  - DEPT $\xrightarrow{?}$ MGR

  - MGR $\xrightarrow{?}$ DEPT

DEPT	PROJECT	MGR
1	A	M1
1	B	M1
1	C	M2
2	A	M3
2	D	M3

Normalization Theory - 1

- Normalization is a matter of semantics
- Semantics are conveyed by dependency information
- Requires good knowledge of the dependency theory
- Conducted step by step in a reversible process

Normalization Theory - 2

- Requirement reasons for normalization
  - Normal form relations are pure structures that do not carry multiple disjoint facts
  - Normal form relations do not cause update anomalies

- Two main design methodologies
  - Decomposition approach
  - Synthesis approach

Normalization Theory - 3

- Types of constraints used in normal forms
  - Constraint on attribute values
  - Constraint on dependency of nonprime attributes on keys
  - Constraint on dependency of prime attributes on all attributes
  - Constraint on MVDs
  - Constraint on projection-join dependencies

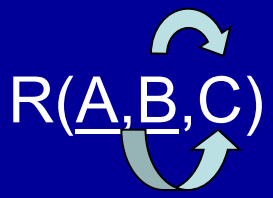
First Normal Form

- 1NF relation must have atomic values for all of its attribute values in every tuple; multiset valued attributes and/or subrelations are not allowed within a 1NF relation

Second Normal Form -1

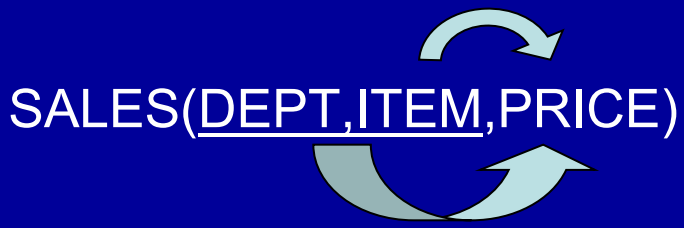
- A **non-2NF** relation;

  - An attribute depends not on the whole but on a portion of a relation's key



Second Normal Form - 2

- Example



- Semantic constraints
  - A department can sell several items
  - A given item can be sold in more than one department
  - All departments charge the same price for the items as company policy

Second Normal Form - 3

- Anomalies in non-2NF

  - Insertion anomaly

    When the company stocks a new item and determines its price, the item and its price can not be entered in the SALES relation until at least one department starts selling it

  - Deletion anomaly

    When a season ends and departments stop selling a given item, then we cannot keep in SALES the fact that the item with its known price exists in our stocks

Second Normal Form - 4

- Anomalies in non-2NF (continued)
  - Update anomaly

    If and when the price of an item changes, multiple tuples in SALES must be updated since several departments may sell the same item. If an update pertains to a single fact, then the update must affect a single tuple not several

Second Normal Form - 5

- To convert a non-2NF to a 2NF, seperate different facts (FDs; AB$\longrightarrow$C, B$\longrightarrow$C) by decomposing non-2NF relation

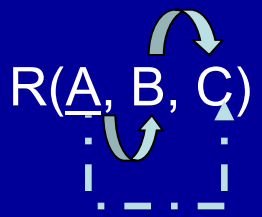
  - Example

    SALES (<u>DEPT, ITEM</u>)

    ITEMINFO (<u>ITEM</u>,PRICE)

Third Normal Form -1

- A **non-3NF** relation;
  - Transitive dependency of nonprime on nonprime



Third Normal Form - 2

- Example

- Semantic constraints
  - ITEM determines its PRICE
  - DISCOUNT amount is determined by the PRICE
  - An ITEM with no DISCOUNT applied would normally have a DISCOUNT amount of zero (and also ITEM is key so F.D : ITEM $\longrightarrow$ DISCOUNT)

Third Normal Form - 3

- Anomalies is non-3NF

  - Insertion anomaly

    Price and discount are determined by company policy and PRICE is enough to determine DISCOUNT. We can not enter the PRICE, DISCOUNT pair unless we have an ITEM with that exact price

  - Deletion anomaly

    If the last item with a given price is sold out, we have to drop PRICE, DISCOUNT pair from the database

Third Normal Form - 4

- Anomalies in non-3NF (continued)

  - Update anomaly

    If the changes policy in determining discount changes, then we have to update multiple tuples in the database for all items bearing the same price

Third Normal Form - 5

- To convert a non-3NF to a 3NF, seperate disjoint facts into seperate relations

Example

ITEMINFO (ITEM, PRICE)
DISCNT (PRICE, DISCOUNT)

E/R Diagrams

- History
  - First published Peter Chen in 1976
  - Extended by several people
    * by Özkarahan in 1986
    * by Doğaç, Özkarahan and Chen in 1988

- Today known as E^2/R diagrams

Introduction

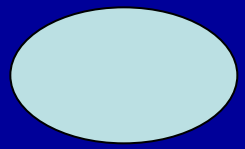
- Structural design aid
- Easy adaptation to network, hierarchical and relational models
- Easy to understand and collaborate on
- Graphical model

Basic Concepts

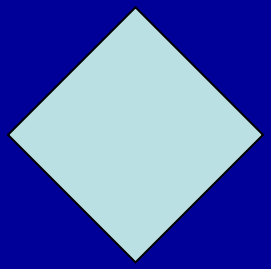
- Entity: An atomic element, record in relational terminology
- Entity set: Denotes an entity type, table in relational terminology
- Property: Attributes of an entity
- Relationship: Relations between the entity sets or is a mapping among the entities of entity sets

Graphical Notation

Entity or Entity Set

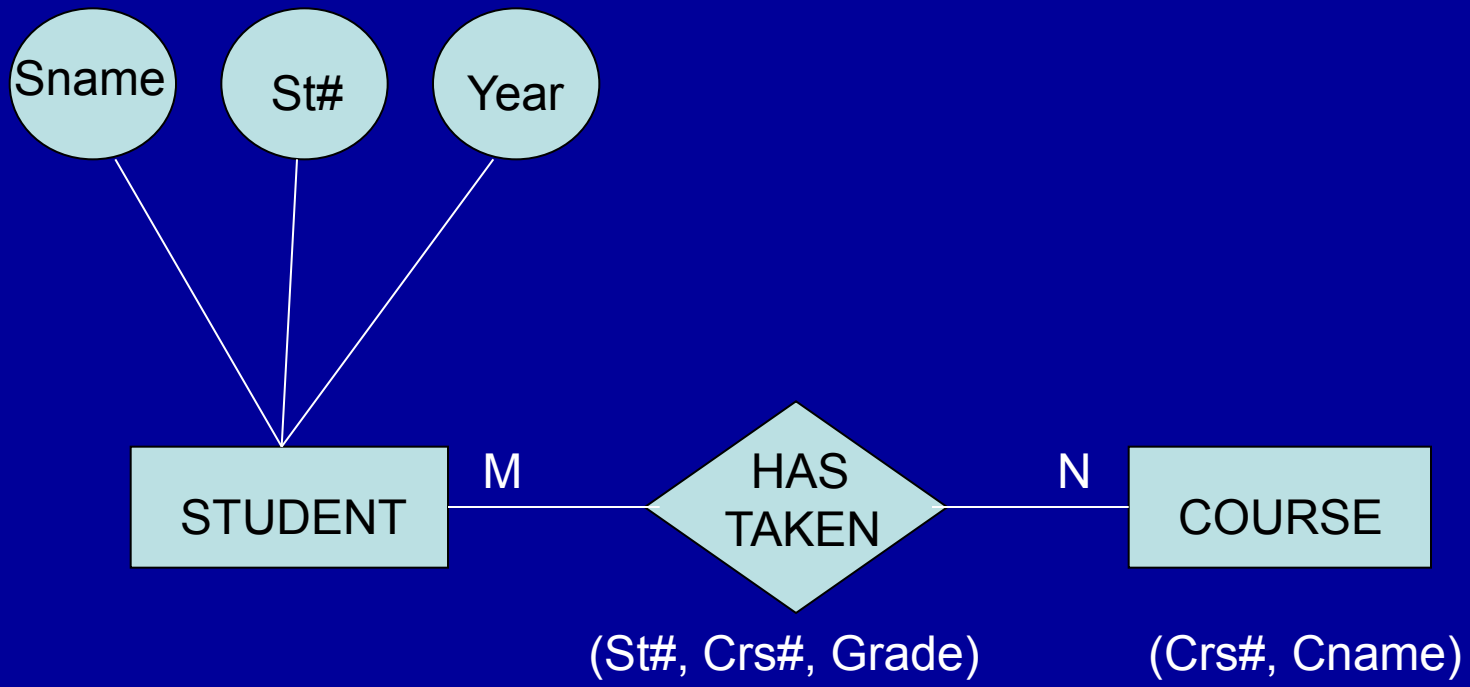


Property

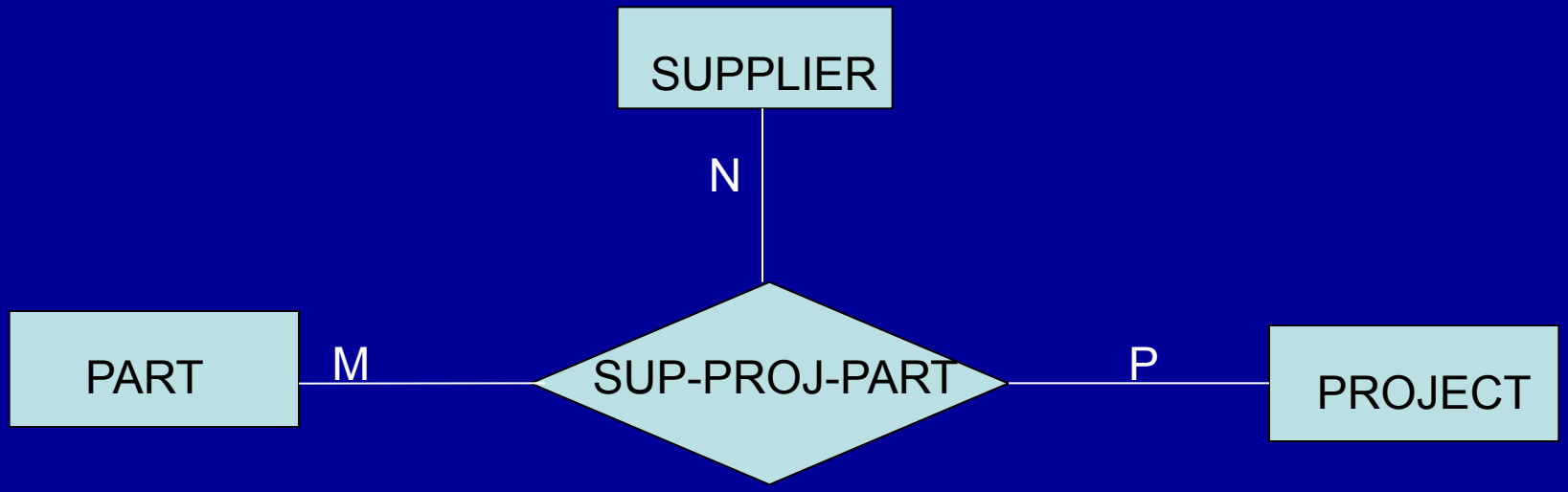


Relationship

Example E/R Diagrams - 1

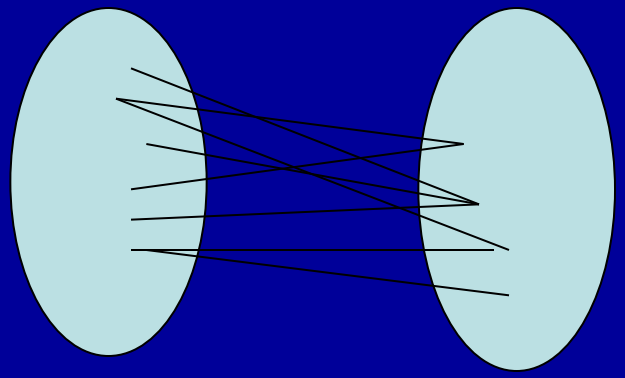


Example E/R Diagram - 2



Mapping Constraints

- 1:1 (One to One , 1-to-1)
- 1:M (One to Many, 1-to-M)
- M:N (Many to Many, M-to-N)
- M:1 (Many to One, M-to-1)



Cardinality Constraints

- (min:max) pair represents a cardinality constraint
- (1:6) means a student can take at least 1 course and at most 6 courses
- (0:40) means a course can not be taken or a course can be taken by at most 40 students

Steps in E/R Modeling

- Identify entities
- Identify relationships among entities
- Identify mapping constraints of relationships
- Identify cardinality constraints of relationships
- Identify properties of all types (entities and relationships)
- Identify keys

Extensions to E/R Model

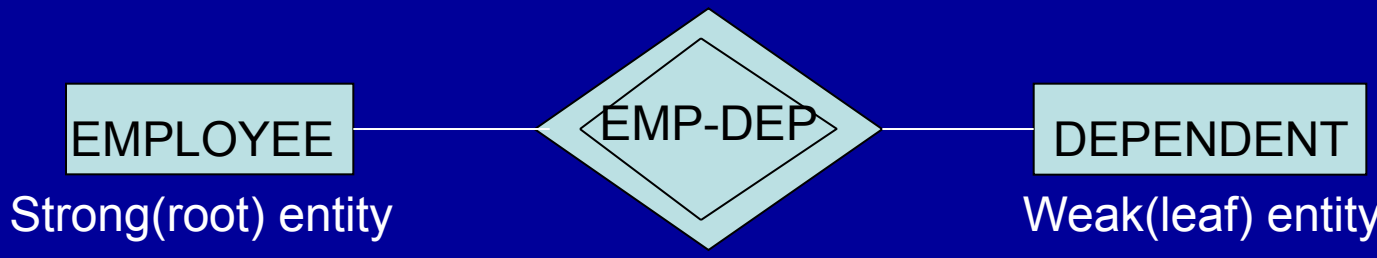
- Weak relationship
- Total relationship
- Aggregation
- Generalization

Weak Relationships-1

- Mapping constraints can be 1:1 or 1:M

  (Ex: Album & Songs, Patient & Prescription)

  DEPENDENT entity can not exist without EMPLOYEE entity



- To represent weak entity and weak relationship
  - double diamond
  - double rectangle

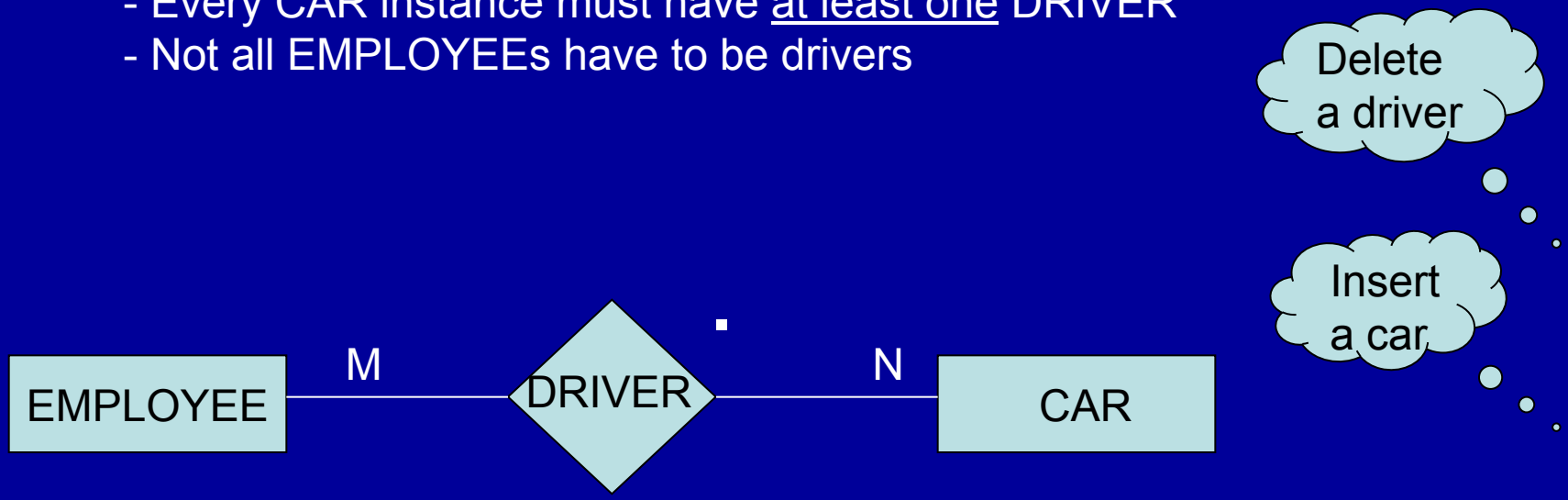
  - rounding the corners of rectangle
  - rounding the corners of diamond

Total Relationships-1

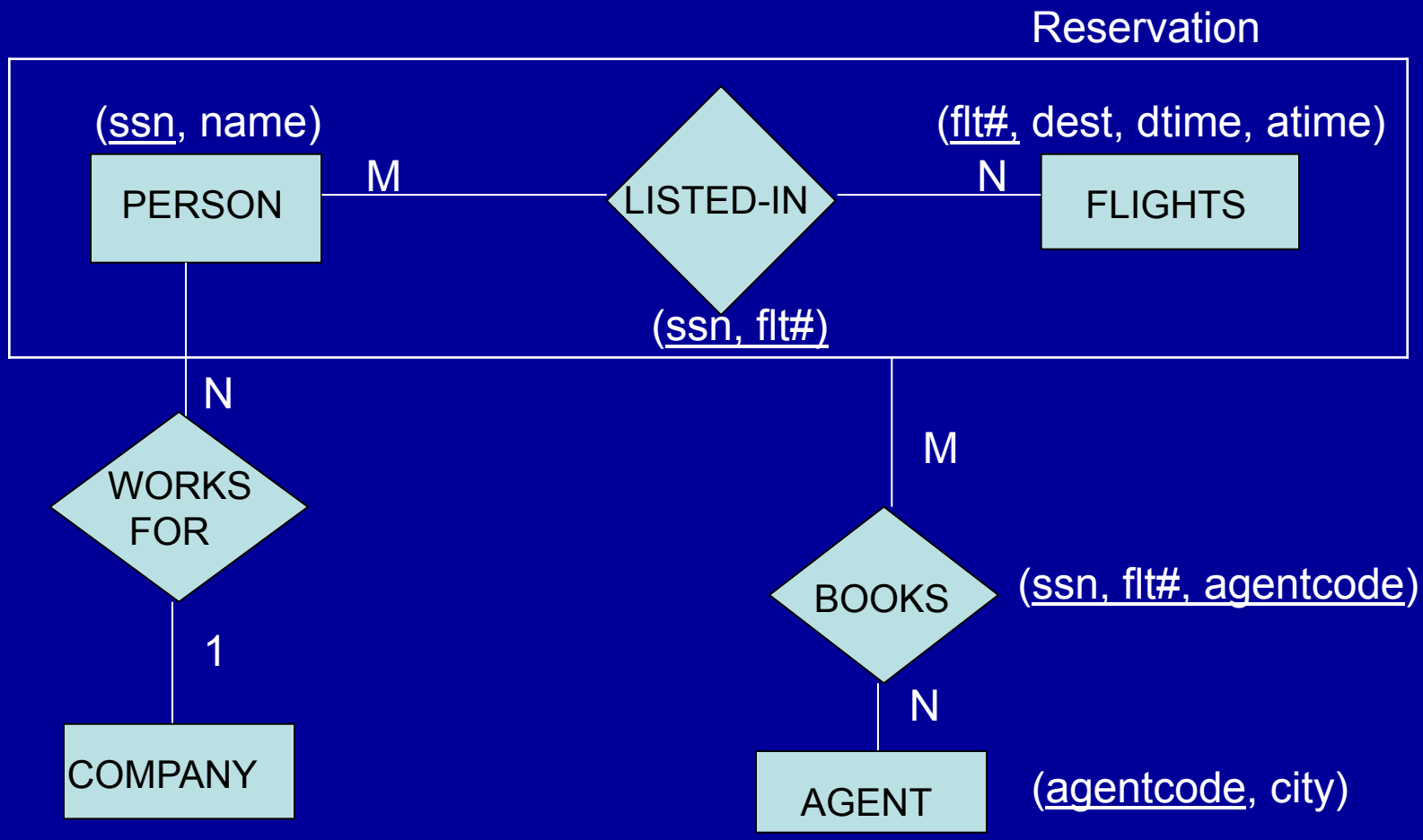
- Mapping constraint can be M:N or 1:M

Example:

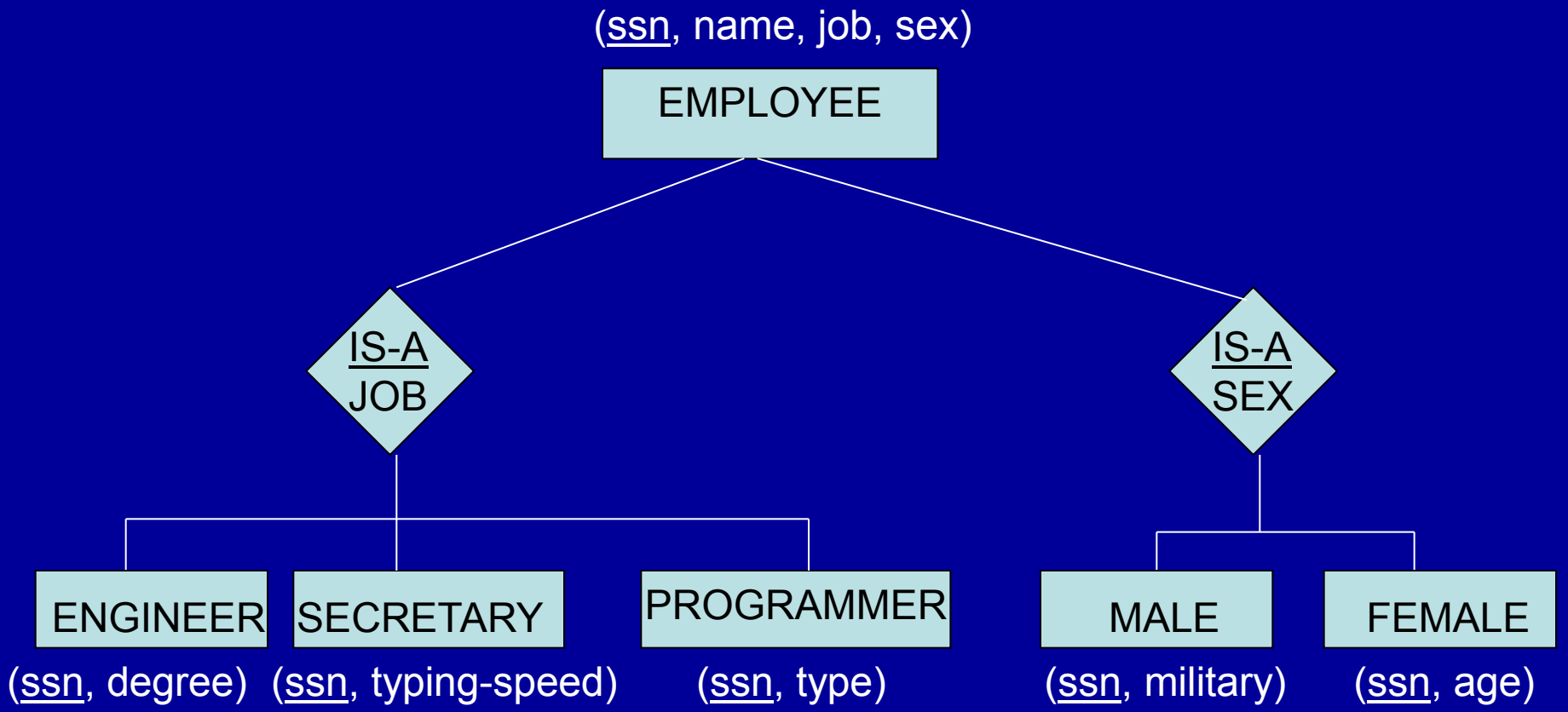
- Every CAR instance must have at least one DRIVER
- Not all EMPLOYEEs have to be drivers



Aggregation



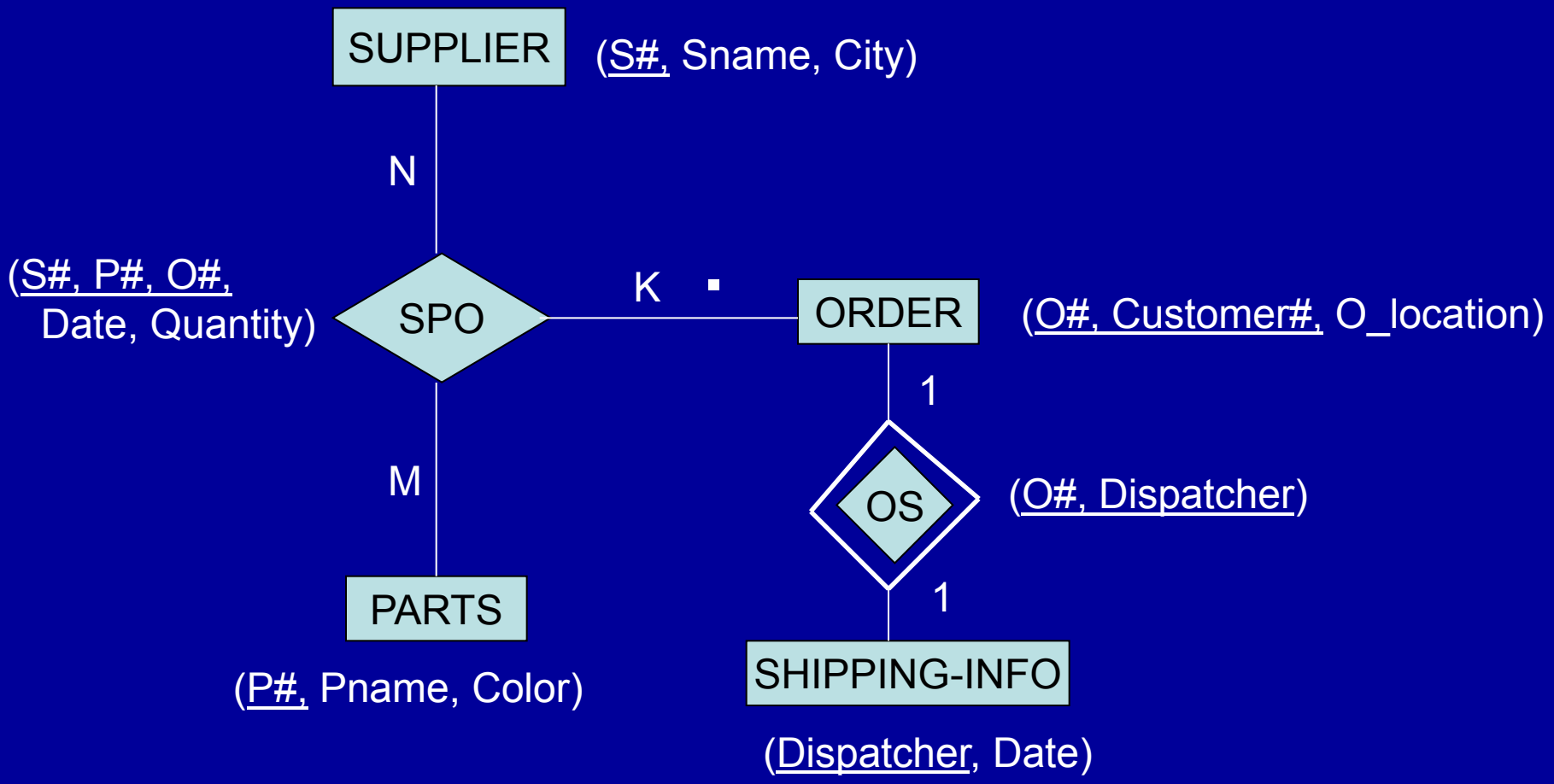
Generalization



Example - PARTS DB

- METADATA

  A certain application in an organization deals with Supplier, Part and Order placed for the parts supplied by the suppliers. There is also piece of information called shipping info whose existence depends on ORDER. S#, Sname and city describe SUPPLIER; P#, Pname and color describe PART. O#, Customer# and O_location describe ORDER; and dispatcher and date describe SHIPPING_INFO. Each supplier can supply several parts to one or more customers. A given part can be ordered by different customer, there can be more than one supplier. There can not be an “inactive” order at a given time. This does not hold for SUPPLIER and PARTS, however. Also, for each order, the date and quantity information should be recorded. Draw an E/R diagram.

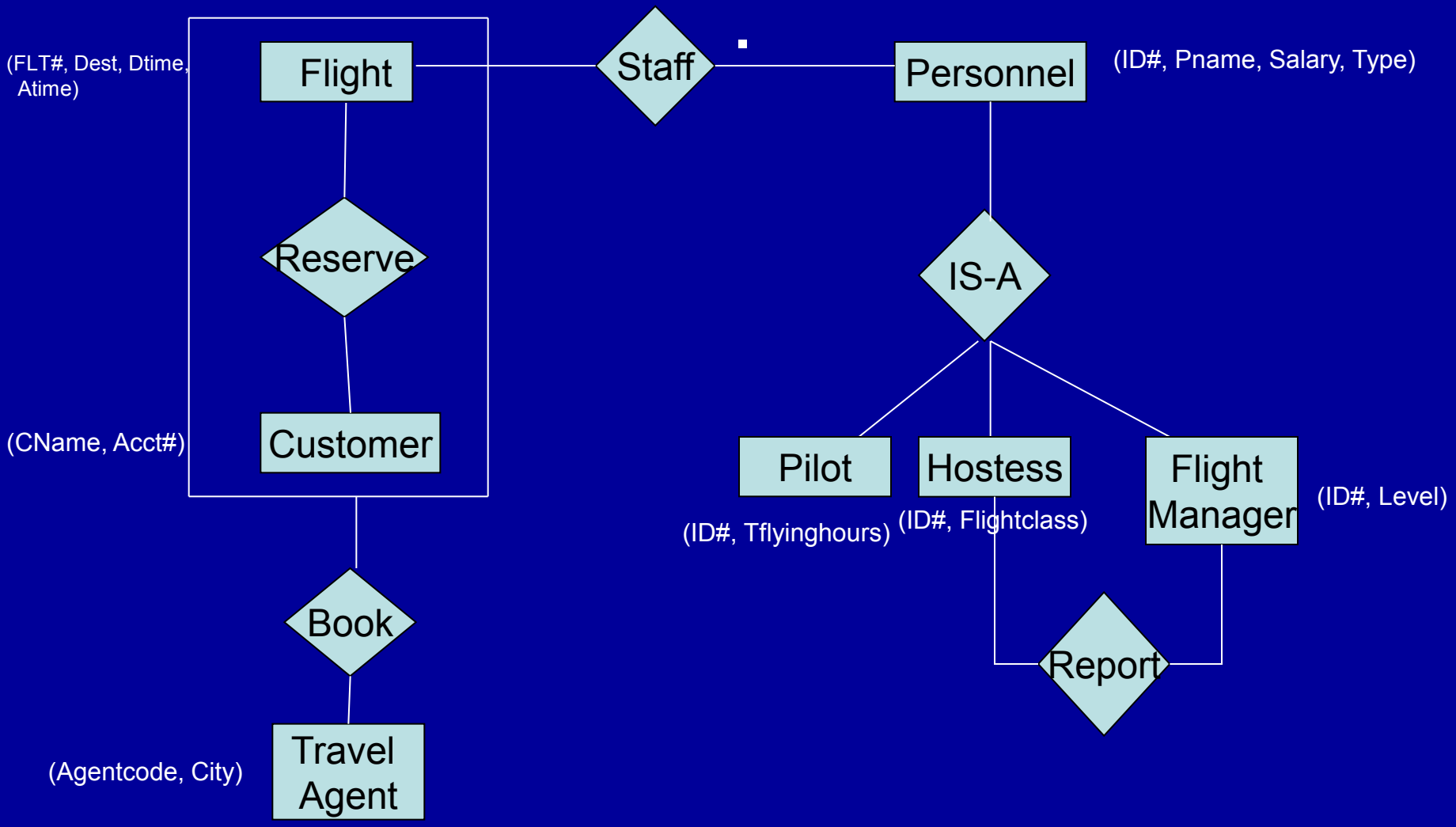

SUPPLIER
(S#, Sname, City)
N
(S#, P#, O#,
Date, Quantity)
SPO
K
ORDER
(O#, Customer#, O_location)
1
M
OS
(O#, Dispatcher)
1
PARTS
(P#, Pname, Color)
SHIPPING-INFO
(Dispatcher, Date)

Example – AIRLINE DB

- Metadata

  Draw an E/R conceptual schema diagram for the following application: In an airlines application, there are flights and personnel staffing each flight. Personnel consists of pilots, hostesses and flight managers. Hostesses report to flight managers. Obviously, there can not be an unstaffed flight. Customers reserve flights and this act is referred to as a “reservation”. Travel agents may book reservations. The following details are available:

  Flight:FLT#, Dest, Dtime, Atime
  Customer: CName, Acct#
  Travel Agent:Agentcode, City
  Personnel:ID#, Pname, Salary, Type
  Pilot:ID#, Tflyinghours
  Hostess:ID#, Flightclass
  Flightmanager:ID#, Level


(FLT#, Dest, Dtime, Atime)
Flight
Staff
Personnel
(ID#, Pname, Salary, Type)
Reserve
IS-A
(CName, Acct#)
Customer
Pilot
Hostess
Flight Manager
(ID#, Level)
(ID#, Tflyinghours)
(ID#, Flightclass)
Book
Report
(Agentcode, City)
Travel Agent

Examples

1) Draw an E/R diagram for the following application: In a university, there are instructors of students advising instructors. There can not be an unadvised student. Instructors teach courses, this act is referred to as a class. Students enroll classes. Each student takes a project and his or her existence depends upon the existence of the project

2) A basketball club has many players(Pno, Pname, Page) and teams (Tno, Tname, Tdirectors). The teams and players are described by their player and team numbers (Pno, Tno). Those players are playing in the teams and it is needed to keep their playing time (playtime) for the team. When a team is created, players can be assigned later to this new team. However, a player must play in at least one team (some players can play in more than one team) otherwise, he is not a member of the club and must be deleted

3) In a library information system three types of item exist. Each item is identified by a unique item id. These items can be purchased to store or can be borrowed by members. A member can borrow more than one item. For purchasing, an item can be stored in many location and a store can have many item. Each member pays an amount of money for each item taken.

Örnekler

Aşağıda verilen örnek senaryolar için E/R diyagramlarını çizin

1) Bir üniversitede öğretim elemanları ve öğrenciler vardır. Öğretim elemanları öğrencilere danışmanlık yapar. Danışmanı olmayan bir öğrenci olamaz. Öğretim elemanları ders verirler ve bu durum sınıf olarak tanımlanır. Öğrenciler sınıflara kayıt olurlar. Her öğrenci bir proje alır ve öğrencilerin var olması projelerin var olmasına bağlıdır.

2) Bir basketbol klubünde çeşitli oyuncular (Pno, Pname, Page) ve takımlar vardır (Tno, Tname, Tdirector) Bu takımlar ve oyuncular oyuncu ve takım numarası ile tanımlanır(Pno, Tno). Takımlarda oynayan oyunculara oynadıkları takımlar için oyun zamanı bilgisi tutulur. Bir takım oluşturulduğu zaman oyuncular bu takımlara daha sonra da atanabilir. Fakat bir oyuncu mutlaka en az bir takımda oynamalıdır (Bazı oyuncular birden fazla takımda da oynayabilir). Aksi halde, o üye klubün bir üyesi olarak kabul edilmediğinden silinmelidir.

3) Bir kütüphane otomasyon sisteminde 3 tip ürün vardır. Her ürün için ürün_id olarak bilinen bir anahtar saha tanımlıdır. Bu ürünler satın alınabilir veya ödünç alınabilir. Bir üye birden fazla ürün ödünç alabilir. Bir ürün birden fazla depoda depolanabilir, bir depoda birden fazla ürün bulunabilir. Her üye aldığı her ürün için belirli bir miktar ücret öder.