# CS 281

Lecture Notes _(Week 2)

Yağmur Yılmaztürk
Gül Çulhan
Anıl Karakaş

10.02.2015

*Setting Up A Database*

1) Define a model containing all appropriate types of data and data relationships.
  Entity relationship model : ER Model

2) Define integrity (correctness) constraints on the data. For example; a dependent can exist with an employee.

3) Define the conceptual schema for the model.
        → Define tables (tables/ relation name attributes)

4) Define the physical level depending on the possible use queries.
        → Define indexes for secondary attributes.

Courses (CourseNo, dept, credit, hour)

Primary attribute   Secondary attributes
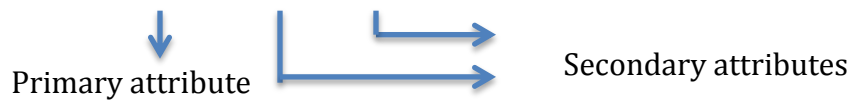
5) Define views

6) Create / Initiate database

7) Give access rights to users

Views:

View 1

View 2

View

Conceptual Level

Physical Level
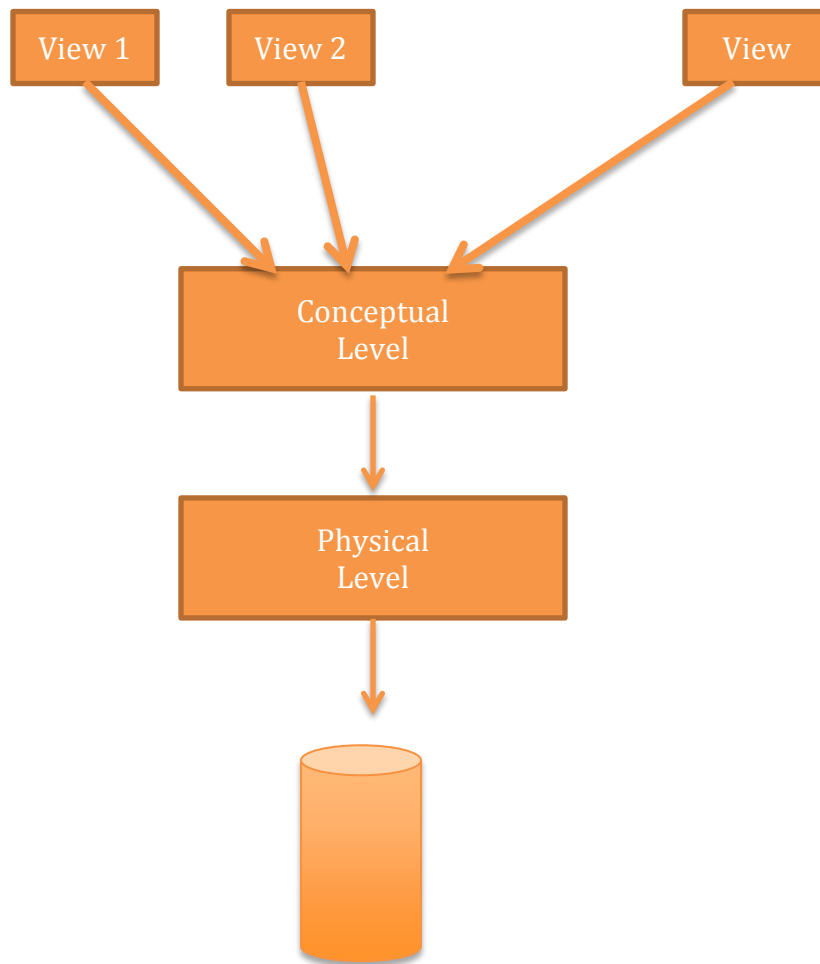
DBA: Database Administrator

Conceptual Schema

Define entities and relationships in terms of tables (relations).

Students (sid: string, fname: string, login: string, age: integer, gpa: real)
Faculty (fid: string, fname: string, salary: real)
Courses (cid: string, cname: string, credit: integer)
Rooms (rno: string, address: string, capacity: integer)
Enrolled (sid: integer, cid: string, grade: string)
Teacher (fid: string, cid: string)

tuples → F10   CS281
      → F10   CS281

MeetsIn (cid: string, rno: string, ctime: string)

F10: There must be a faculty corresponding to F10 in faculty relation.

reference relation → referential integrity

View: External Schema Example

CourseInfo (cid: string, fname: string, classSize: integer)

This table is not stored but generated and used

Entity Relationship Model

Requirement analysis

Draw ER diagram

Map ER diagram to relation schemas

Real World

Composed to entities and relationships

Entities

Real objects/ entities

students

faculty members

Conceptual entities

course

bank account

feature =attribute

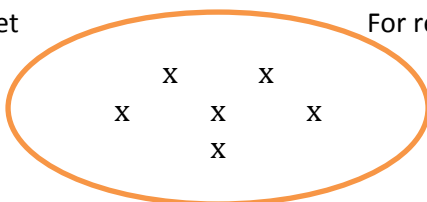Student

student No
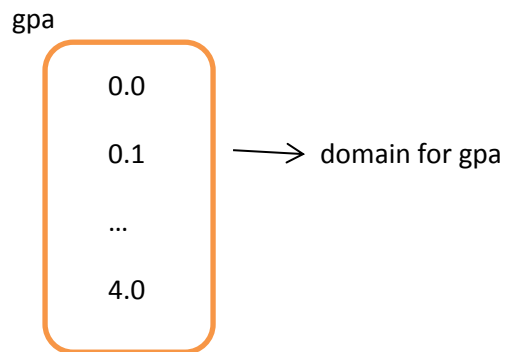
student Name

GPA   Attributes

address

…

students

set   For relationships we have corresponding relationship sets.

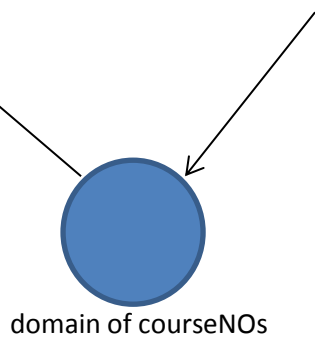Attributes assume/ get values from a corresponding set (domain)

gpa

```
0.0

0.1          ───────→   domain for gpa

…

4.0
```

Employee

salary ───────┐
              │
eno   ────┴──→   domain   1000<salary<20,000

…

Faculty (fNO, courseGiven)   Course(cNo, cName, credit)
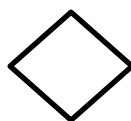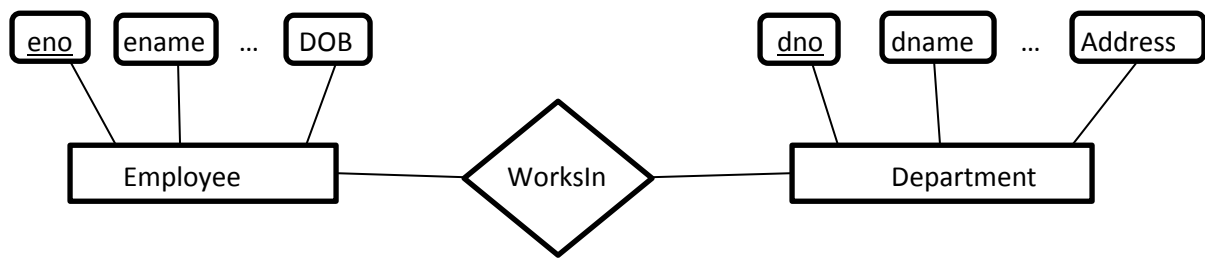
cs 281

domain of courseNOs

Domains are important for enforcing integrity (correctness) constraints.
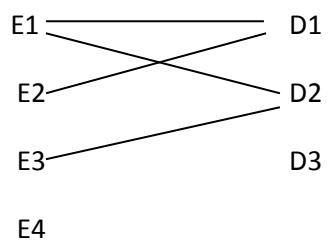
entity   ┌──────────┐
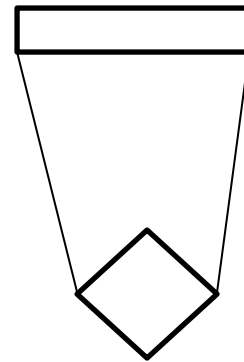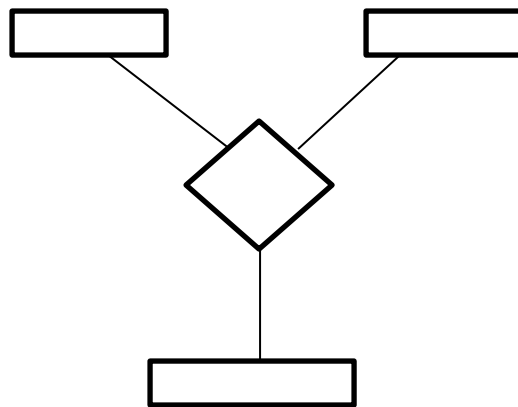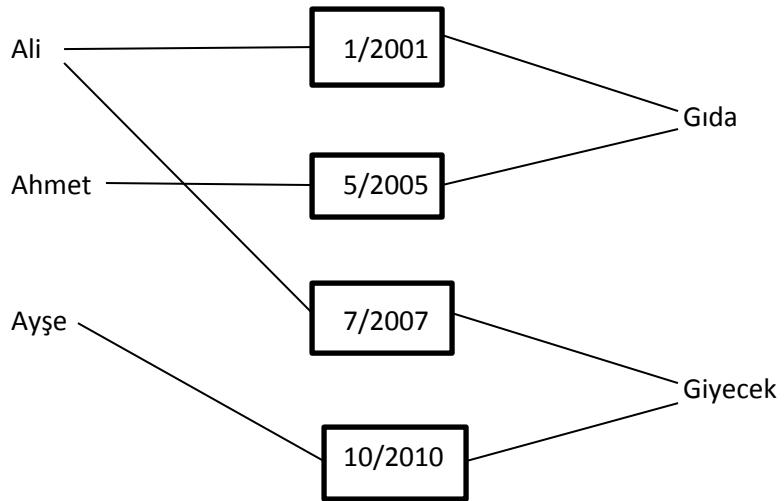   └──────────┘

relationship   ◇

Binary relationship involves two entities

Tenary -3 entities

E1 ——————— D1

E2 ———— D2

E3 ———— D3
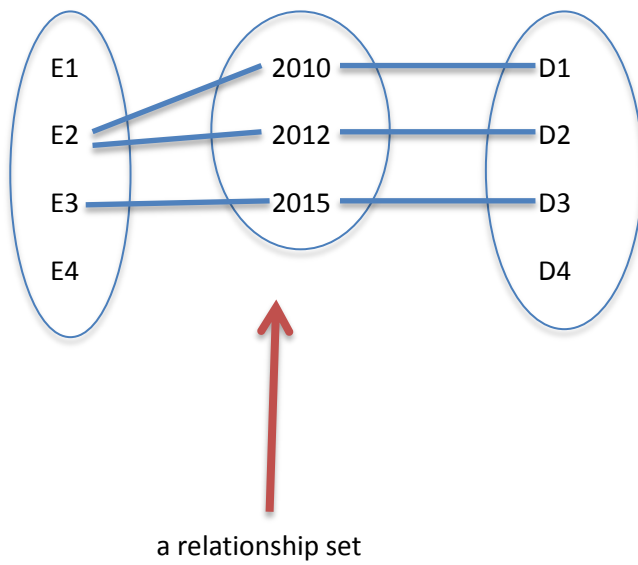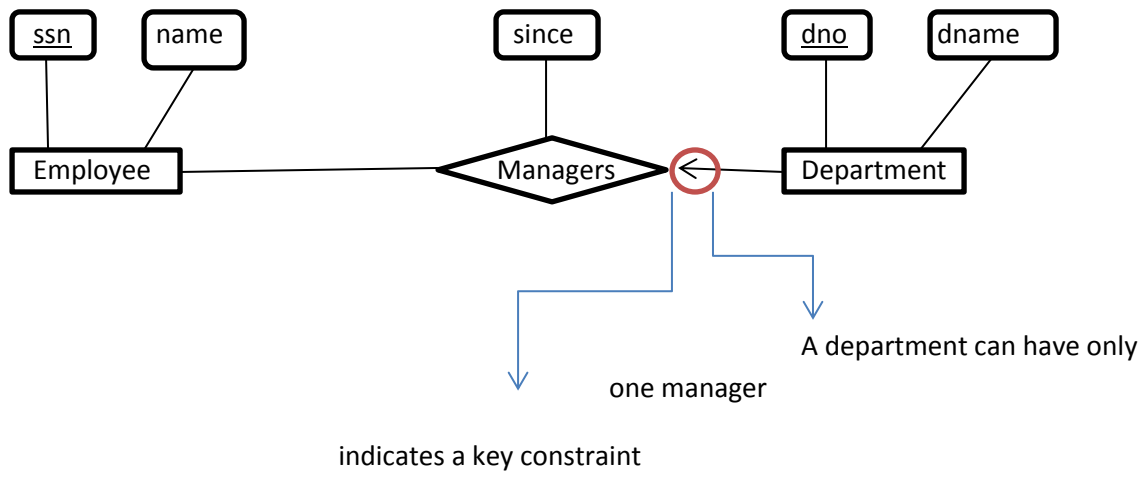
E4

An example for relationship set.

Ternary Relationship Example

Key Constraint

A department can have only one manager

indicates a key constraint

a relationship set

Employee — ◇ Managers ◇ ← Departments

from — to

Doctor   Test

◇ DTP ◇

Nurse   Patient — ◇ Yakın ◇ — H.Yakını
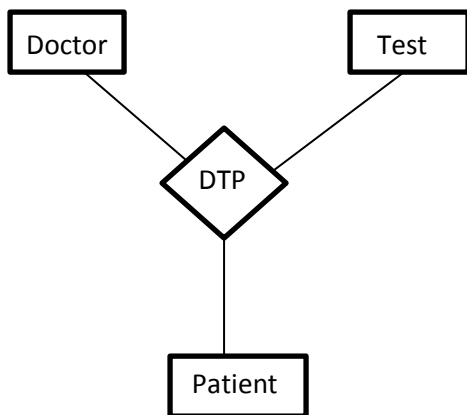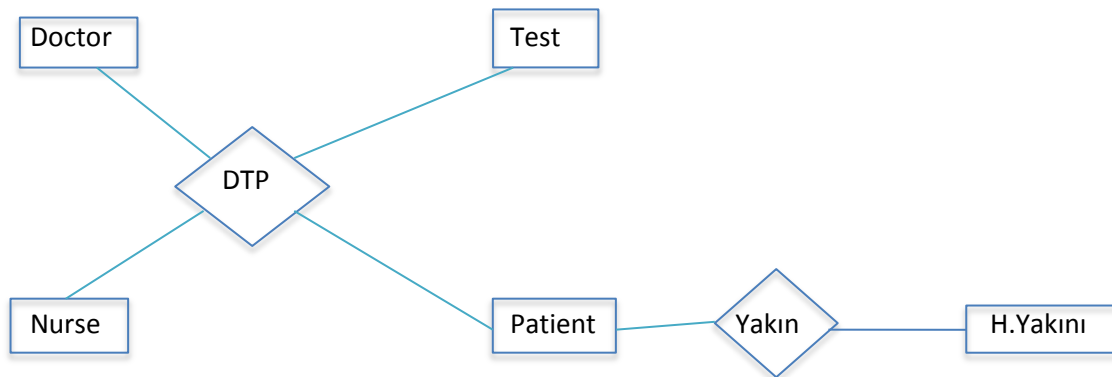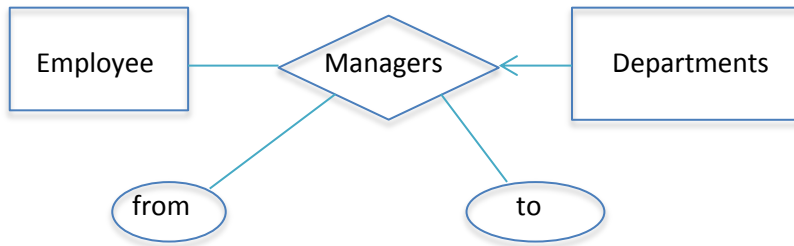
——————— -partial participation

━━━━━━━ -total participation

E ———— ◇ ———— D   employees manage departments, departments
                     managed by employees

E ———— ◇ ━━━━ D   for departments we have one or more
                     managers

E ━━━━ ◇ ———— D

E ━━━━ ◇ ━━━━ D

Employee — Manager — Department

since

WorksIn

since

key constraint