

COMP1531

Week 10 Tutorial!

Last tut!

- demo today
- shared notes for exam prep: <https://bit.ly/2IAOSIM>

- MyExperience is out, please give feedback :)



Quick Recap with ER

Components of ER diagram

- **Entity** - basically a thing of interest
- **Attribute** - a characteristic or property of interest
- **Entity set** - set of entities with the same attributes (analogous to *class* definition in OO)
- **Key** - consist attribute that is unique over the entity set

Relationships in ER Diagram

- relates two or more entities
 - student <is enrolled in> course
- **degree** of a relationship is the number of participating entities, mostly binary (degree = 2)
- **cardinality** - number of associated entities on each side of the relationship
 - one-to-one
 - one-to-many
 - many-to-many
 - *give example of manager, branch, account and customer*

Cardinality

one-to-one



one-to-many

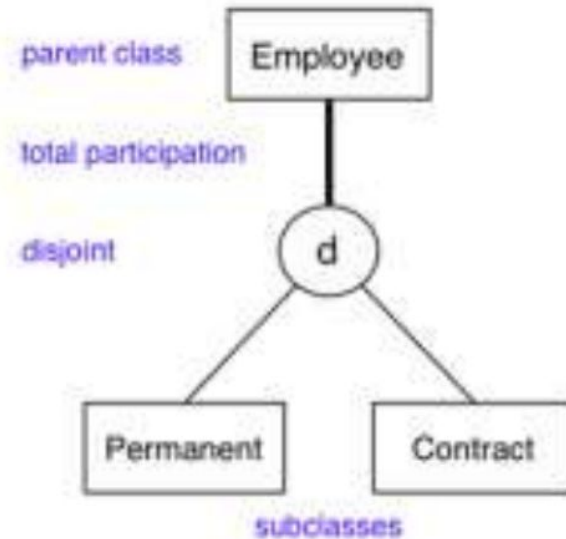
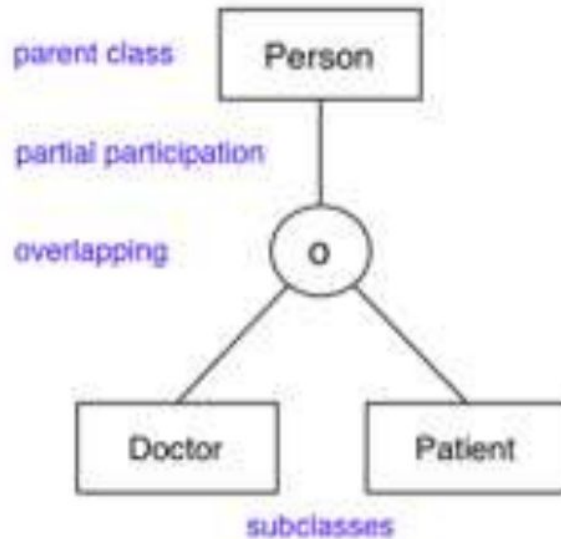


many-to-many



More ER

Disjoint or Overlap



Answer

1. A Person may be a Doctor or a Patient or both ((o)verlapping).

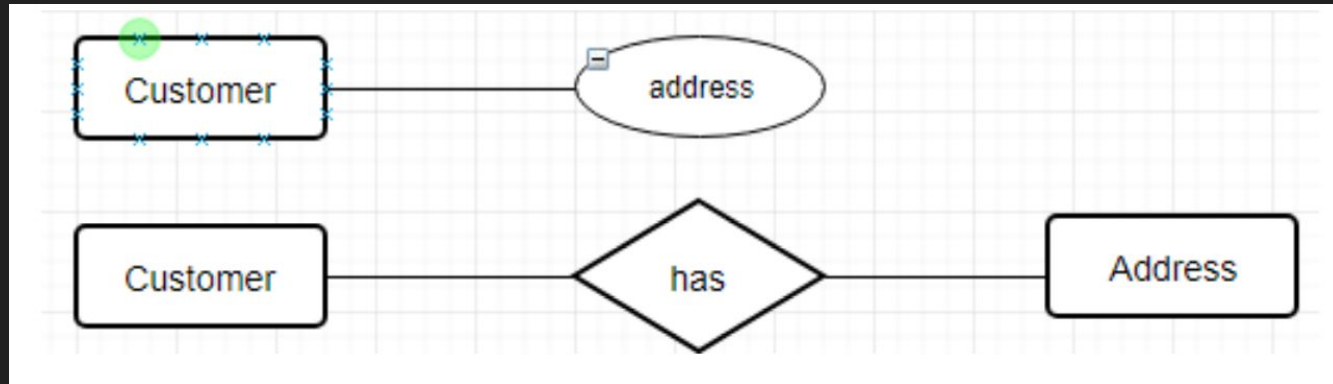
The thinline from Person shows partial participation which suggests that there could be a person who is neither a doctor or a patient

2. An Employee could be either Permanent or Contractor but not both ((d)isjoint).

The thick line from Employee indicates “total participation” which suggests that an Employee must either a Permanent or a Contractor

Which one is better?

The following two diagrams both represent that a customer has an address. Why might we favour one over the other?



Answer

We want the second one.

Why: The first ER diagram shows that a Customer has only one address, The second ER diagram is useful to model a scenario, where a Customer may have multiple addresses such as a billing address, delivery address. Also, the second design could also easily accommodate history information, by adding attributes such as start-date and end-date to the has relationship to show that a Customer has multiple billing addresses, at different periods of time.

Self-paced Exercise

Produce an ER design for a university administration database that records information about faculties, schools, lecturers, students, courses, classes, buildings, rooms, marks. The model needs to include:

- for each faculty, its name, its schools and its dean. Every faculty has one dean
- for each school, its name, the location of its school office, its head and its academic staff; every school has one head
- for each lecturer, their names, birth-date, position, staff number, school, office, the courses they have convened, and the classes they have run
- for each student, their names, birth-date, student number, degree enrolled in, courses studied, and marks for each course. Every student must be studying one or more courses

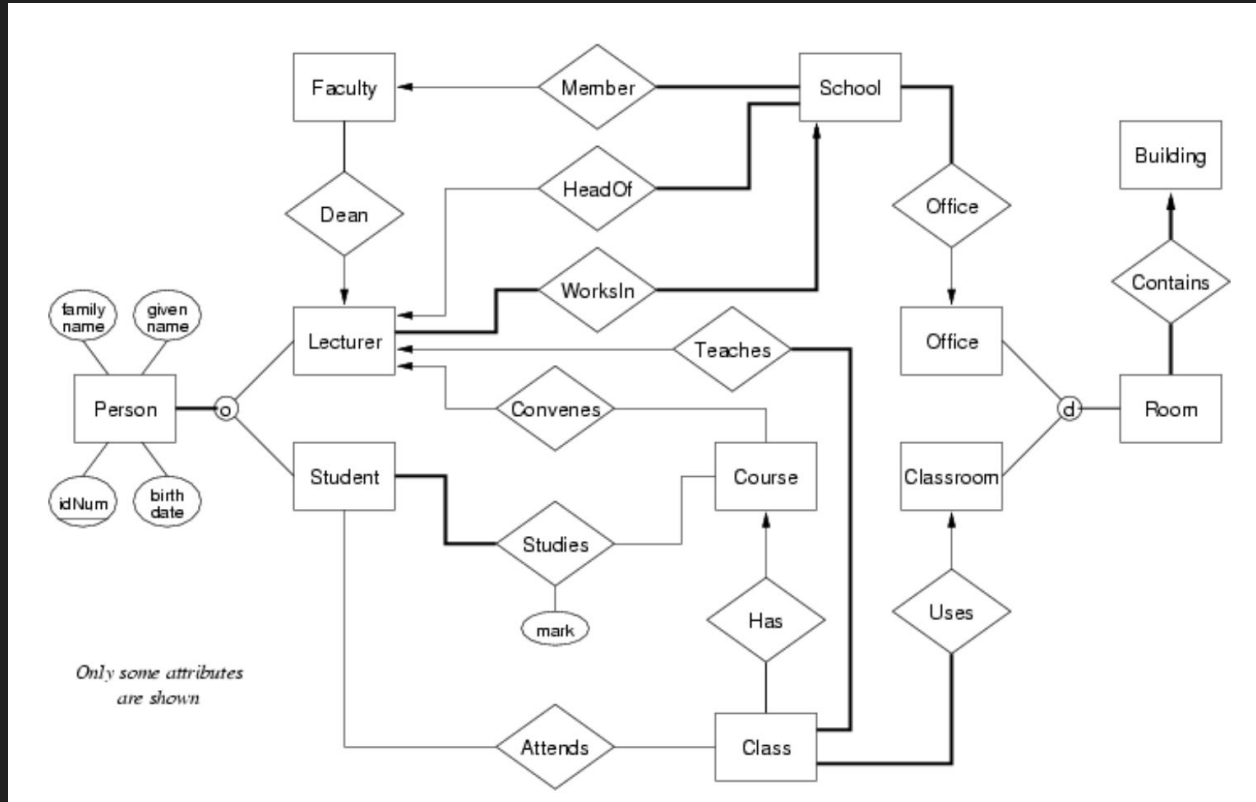
Self-paced Exercise (Cont.)

- for each course, its code, its name, the session it was offered, its lecturer(s), its students, its classes.
- Not all courses necessarily have classes (e.g. reading courses)
- for each class, what kind of class (lecture, tutorial, lab class, ...), its day and time (starting and finishing), who teaches it, which students attend it, where it's held. every class is scheduled in one classroom and is associated with one course every class has one teacher
- for each building, its name and map reference
- for each room, its name, its capacity, type of room (office, lecture theatre, tutorial room, laboratory, ...) and the building where it is located

(a) Produce an ER design for the above database. State all assumptions made.

(b) What cannot be modelled with an ER design?

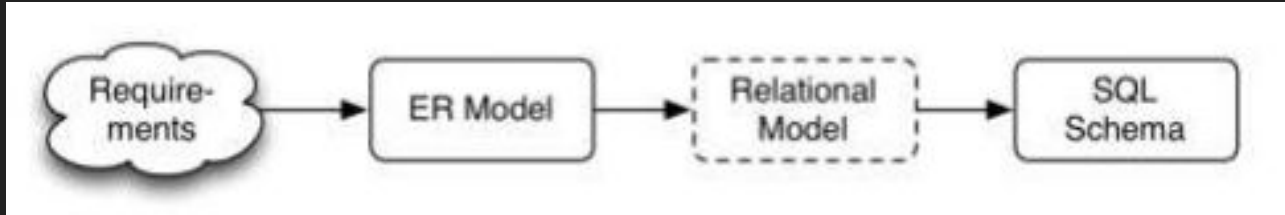
Self-paced Exercise - Answer



Relational Model

Relational Model

- Mapping from ER to Relational model



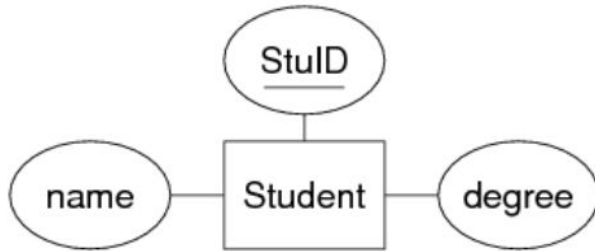
- collections of tables

Relational Model

- express entities and their relationships using just regular tables of data
- **relation** - table of attributes
- entity becomes **relation** (table)
- attribute becomes **field** of the relation
- Unique attribute of the entity -> **Primary Key**

Example

ER Model



Relational Version

Student

<u>StuID</u>	name	degree
--------------	------	--------

Cardinality in Relational Model

- 1 - M relationships in the ER: add a foreign key attribute to the M side
- 1 - 1 Relationship in ER: choose either side to add foreign key (if there's only one side participates totally, choose that side)
- M - M relationships: add whole new relation containing the two foreign keys
- + Draw arrow from foreign key to primary key

Example

- Teacher teaches multiple classes
- Each class has many students
- Each student enrolls into many classes

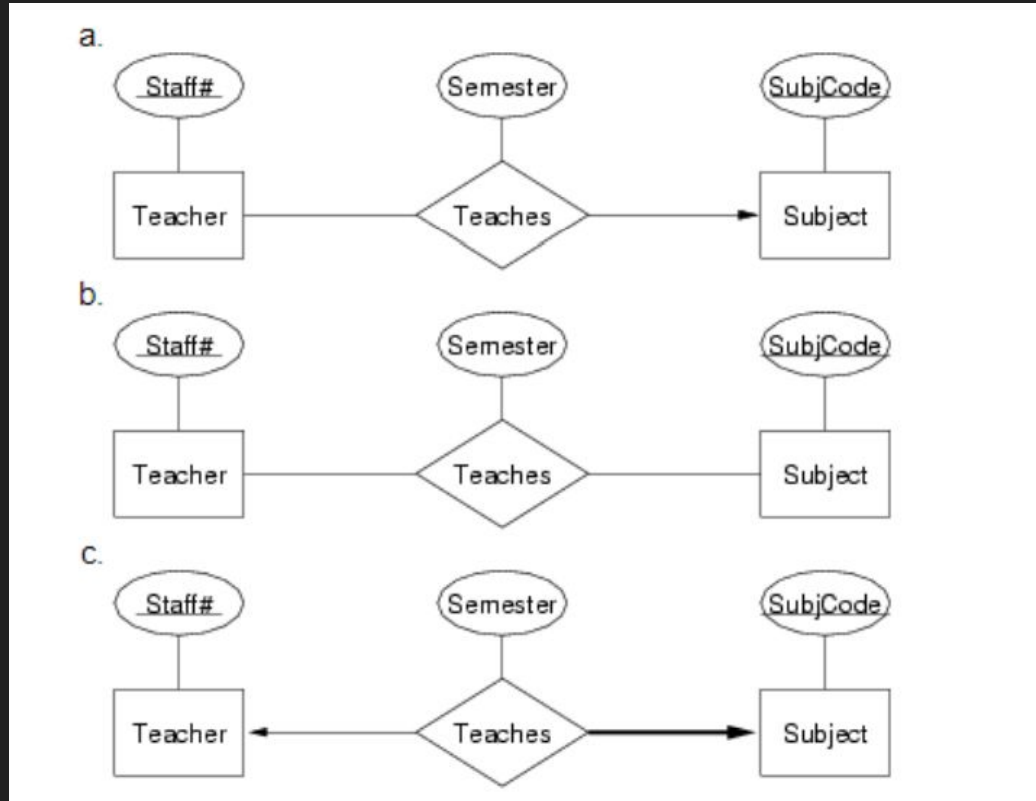
(Workout on the board)

Attribute on the relationship in ER

Adding “semester” in the previous example

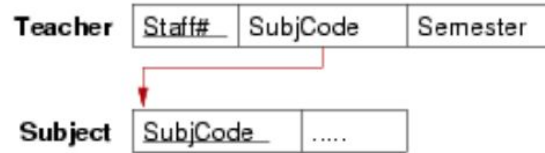
- 1 - M relationships - put on the side that you added foreign key (M side)
- M - M relationships - put the attribute in the new table that contains the foreign key.

More examples

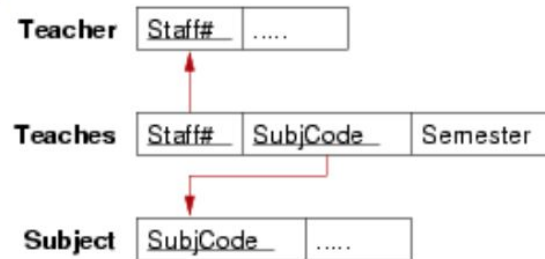


More examples - Answer

a.



b.



c.

