1 Regular Entities

- make a table
- simple attributes \rightarrow attributes/columns
- ullet composite attributes ullet just the components become columns
- choose a key to be the primary key
- uniqueness constraints on other keys
- do not include: derived and multivalued

2 Weak Entities

- make a table
- all simple attributes \rightarrow attributes
- components of composite \rightarrow attributes
- add primary key from owning entity as attribute
- make combination of partial and owning entity key \rightarrow primary key
- foreign key from borrowed key back to key in owning entity

3 1:1 Relationships

Let R be a 1:1 relationship between entities S and T.

3 choices depending on participation

- (a) S has full participation in R, take primary key of T and add it as attribute to S. Make borrowed attribute a foreign key back to T.
- (b) If both S and T fully participate in R: S and T can merge into one table. (Option (a) is still preferred.)
- (c) Not recommended unless there is **partial** participation on both sides. Add a cross-reference table. (Expanded further in step 5.)

4 1:N Relationships

Let R be a 1:N relationship between S and T

2 choices depending on participation

- (a) Assuming T has full participation in R (most common case), then take primary key from S, add to T, and make foreign key back to S.
- (b) (not preferred, unless T has partial participation) cross reference table

5 N:M Relationships

Let R be an N:M relationship between S and T.

- create a new table (called a cross reference table)
- add as columns: primary key attribute(s) of S, T
- make each "borrowed" key a foreign key reference back to origin
- make all attributes primary key for new table

Example: using xref table:

6 Multi-valued Attributes

- treat like weak entity
- create table containing:
 - multivalued attribute
 - primary key of the entity containing attribute
 - whole thing is primary key
 - make column with the primary key of containing entity a foreign key back to the containing table

7 N-ary Relationships

• x-reference table using the primary key attributes of the participating tables

8 Relational Schema Derived from Above Algorithm

- course (course_id, title, hours) (1)
- ullet instructor (name, office, email) (1)
- section (course_id, section_id, instructor_name)
 (2, 4)
 - FK from section.course_id to course_course_id
 - FK from section.instructor_name to instructor.name
- instructor_department_xref(instructor_name, department_name) (5)
- section_meetings(course_id, section_id, days, time, room) (6)
 - whole thing is a primary key
 - foreign key on course_id, section_id

9 Example of multiple valued primary key

```
CREATE TABLE ...
columnname type,
...
primary key (col1, col2),
foreign key (col1) references ...
```