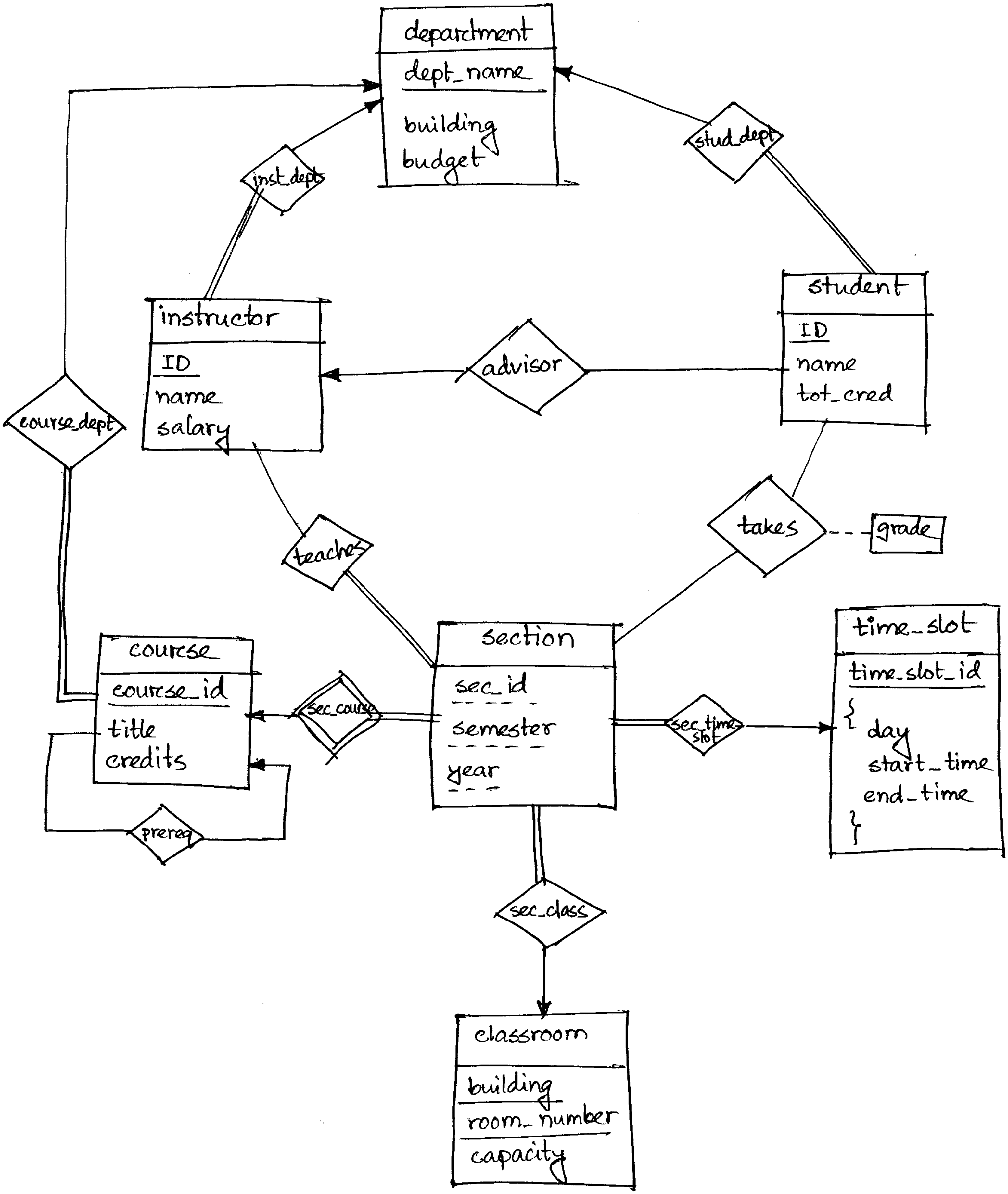
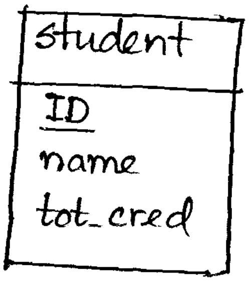

department
dept_name
building
budget
inst_dept
stud_dept
instructor
ID
name
salary
advisor
student
ID
name
tot_cred
course_dept
teaches
takes
grade
course
course_id
title
credits
sec_course
section
sec_id
semester
year
sec_time
slot
time_slot
time_slot_id
{
day
start_time
end_time
}
prereq
sec_class
classroom
building
room_number
capacity

Basic Rules

* Both Entity sets and Relationships becom relations (tables in relational DBMS)

(1) Strong Entity Sets with Simple Attributes

* primary key of entity set → primary key of schem
* Each tuple is one entity
* Each tuple is composed of the entity's attributes



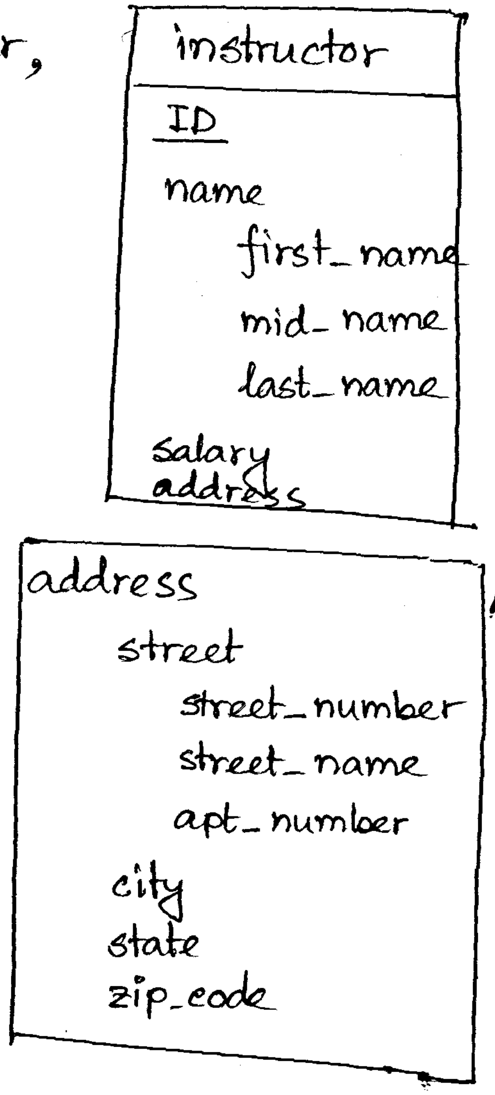
→ student (ID, name, tot_cred)

similarly,

department (dept_name, building, budget)
instructor (ID, name, salary)
student (ID, name, total-cred)
courrse (courrse_id, title, credits)
classroom (building, room-number, capacity

② Strong Entity Sets with Complex Attributes

consider,



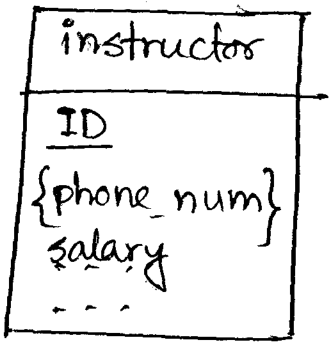
attribute

Case-1

composite attribute

→ instructor (ID, first_name, mid_name, last_name, salary,
{ street_number, street_name, apt_number, city, state, zip_code)

Case-2 (Multi-valued attributes)

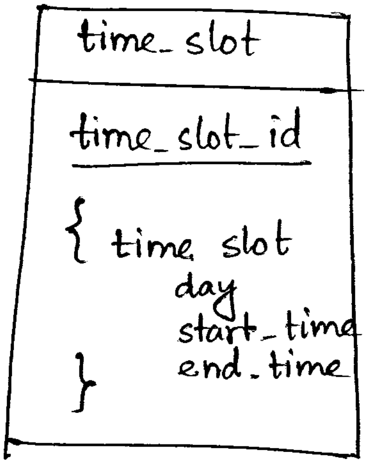


→ instructor (ID, salary, - - -)

instructor_phone (ID, phone_num)

(PK)

special case : only primary key and a multi-valued attributes in the entity set .



→

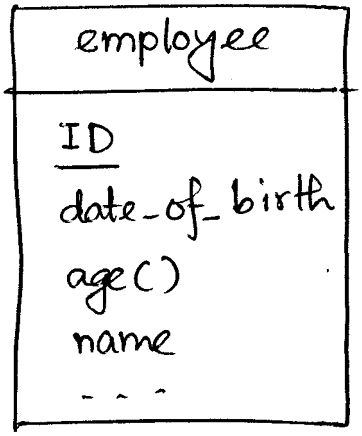
Redundant / Doesn't make sens

time_slot (time_slot_id)

time_slot_id (time_slot_id, day, start_time, end_time)

Case-3 (Derived Attribute)

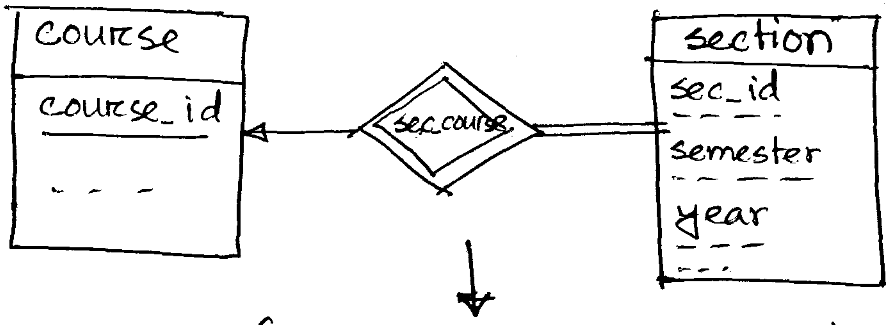
Not included in relational schema .



→ employee (ID, date_of_birth, name . . .)

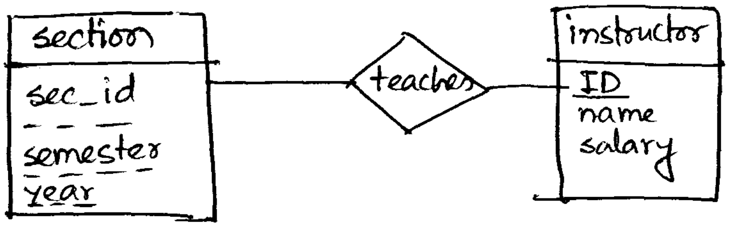
③ Weak Entity Sets

- * Add the primary key of identifying entity set to attributes
- * discriminator + primary key of identifying entity set ⟶ new primary key.



section (course_id, sec_id, semester, year, - - -)

④ Representing Relationship sets ~~(Many-to~~

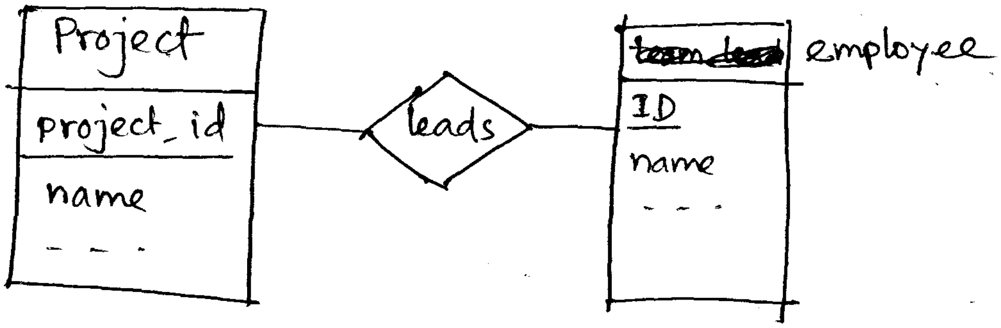


teaches (ins_id, course_id, sec_id, semester, year)

similar takes (student_id, course_id, sec_id semester, year, grade)

Case-1 : Binary, Many-to-Many

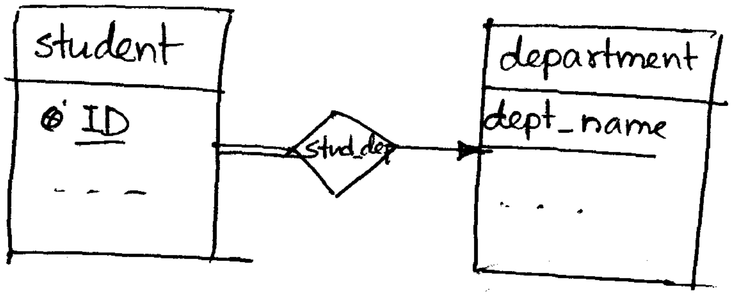
Case-2 Binary one-to-one :-



leads(project_id, employee_id)

leads (project_id, employee_id) is also correct .

Case – 3 Binary many-to-one /one-to-many :-

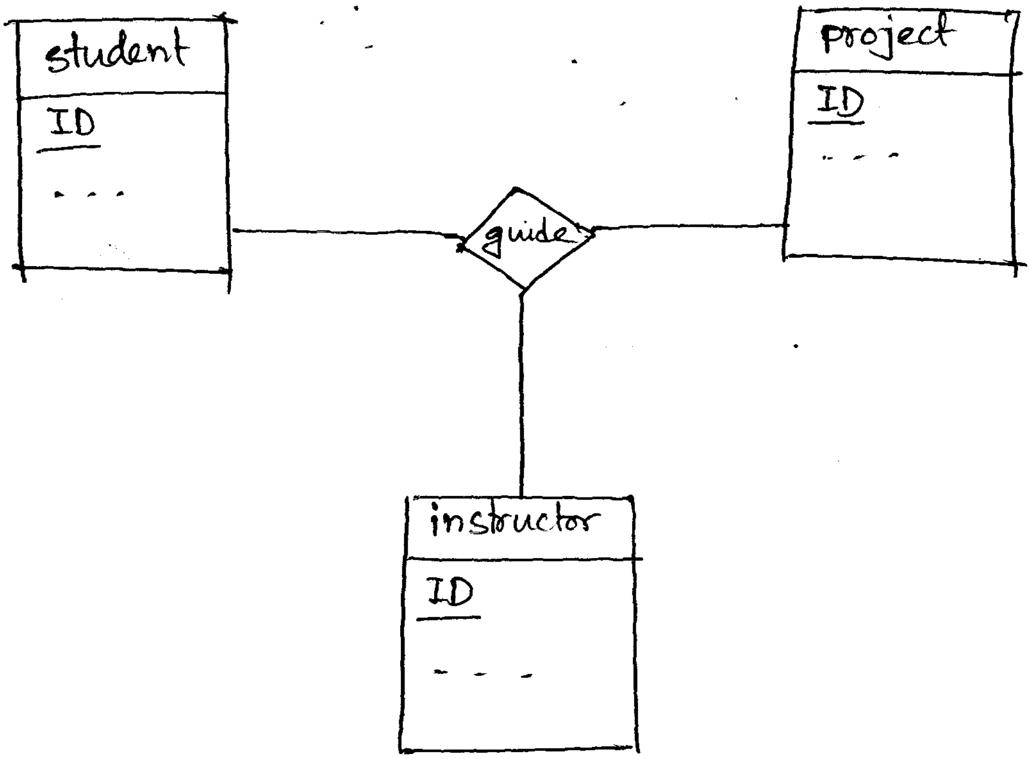


stud_dep (student_id, dept_name) ✓

stud_dep (student_id , dept_nai

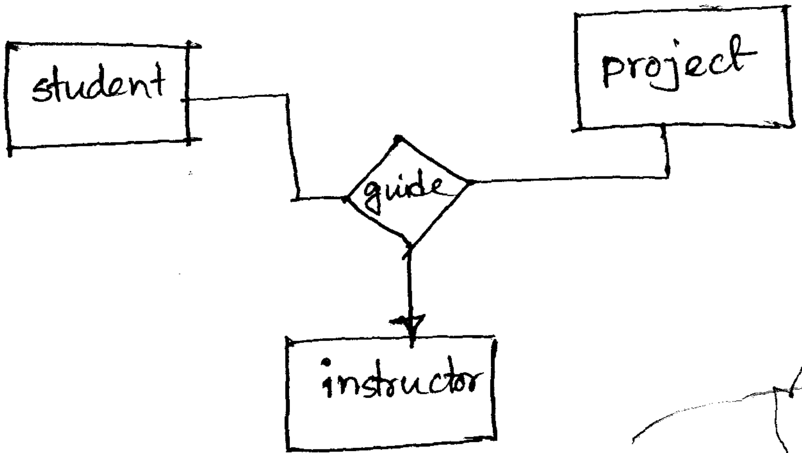
✗

Case-4 n-ary without any arrow :-



guide (student_id, proj-id, ins-id)

case-5 n-ary with arrow :-



guide(student_id, proj_id, ins_id)

excluded as part of primary key