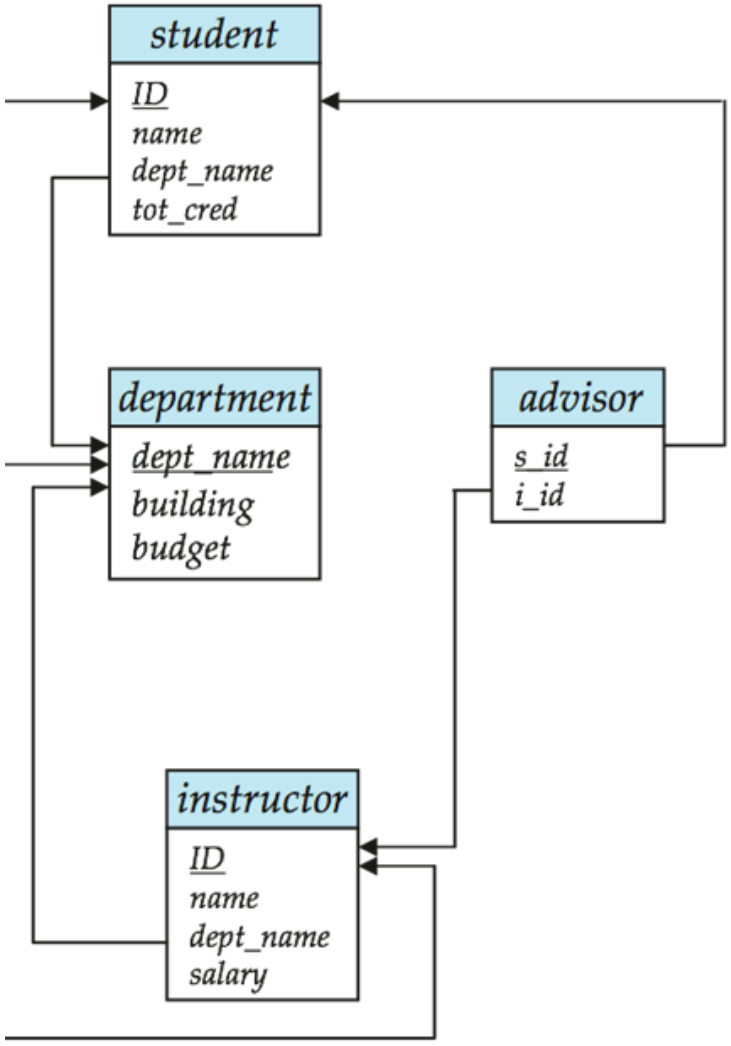
CHAPTER 3
DISCUSSIONS 2

Discussion 3-6

Represent the following query in SQL.

- Find instructors whose name starts with ‘E’ and ends with an ‘n’.

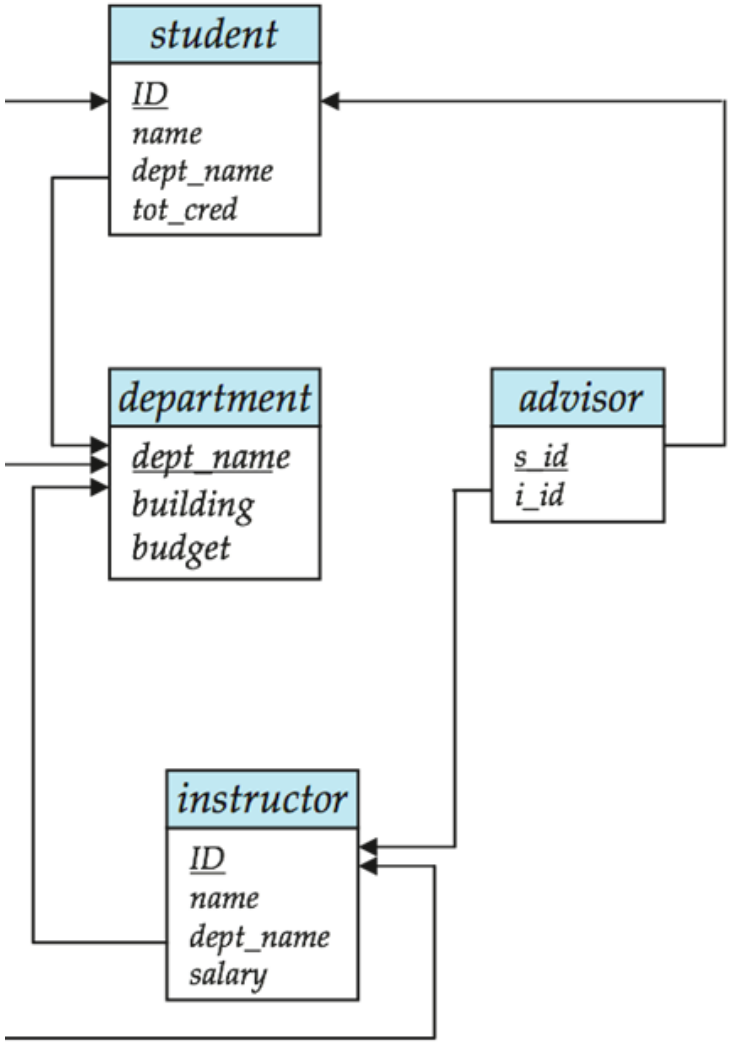


Discussion 3-7

Represent the following query in SQL.

A. Find instructor (ID and name) and student pairs (ID and name) where the instructor is the advisor of the student.

B. Rename the columns of the output of the previous query as *Adv_ID, Adv_Name, Stu_ID, Stu_Name*

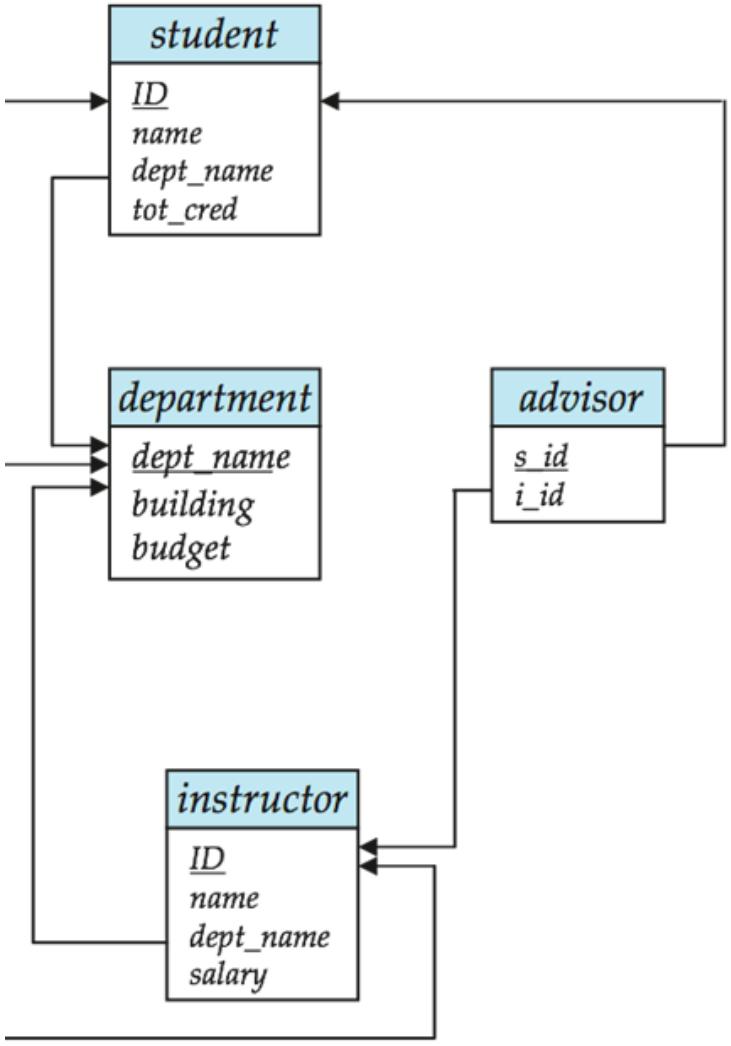


Discussion 3-8

Represent the following query in SQL.

A. Find name of students who has an advisor

B. Find ID of students who do not have an advisor

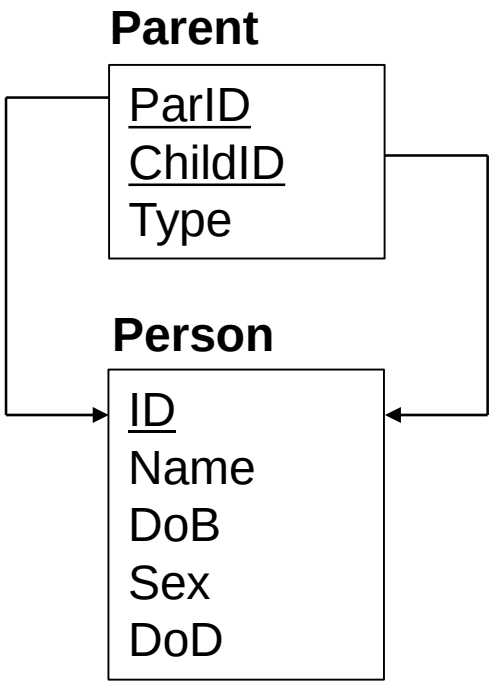


Discussion 3-9

Represent the following queries in SQL.

A. Find the grand children of James Kim (ID: 103999).

B. Find the descendants of James Kim (ID: 103999).

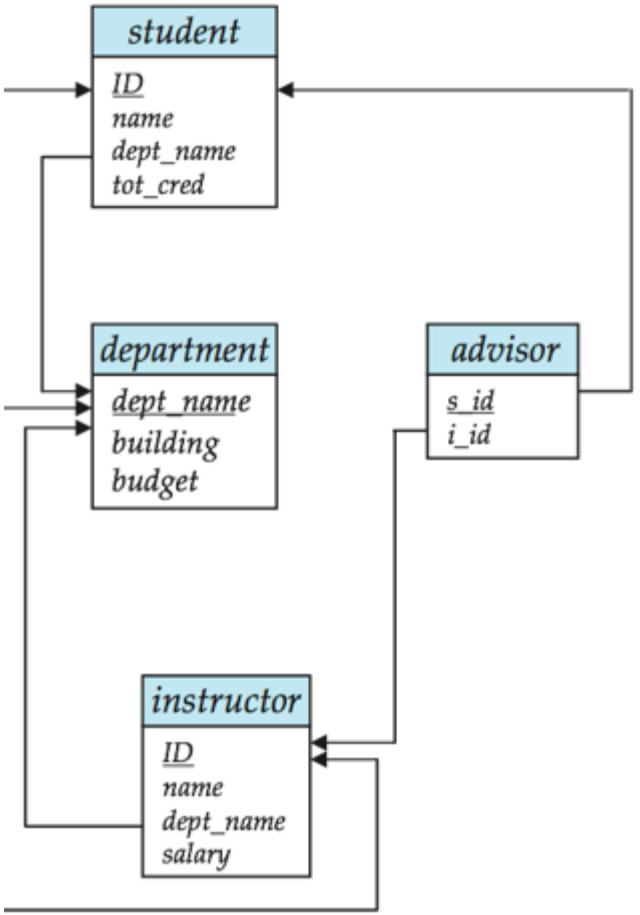


Discussion 3-10

- Represent the following query in SQL.

A. Minimum, maximum, and average budget of departments.

B. Number of advisees of Prof. ‘Kim’.

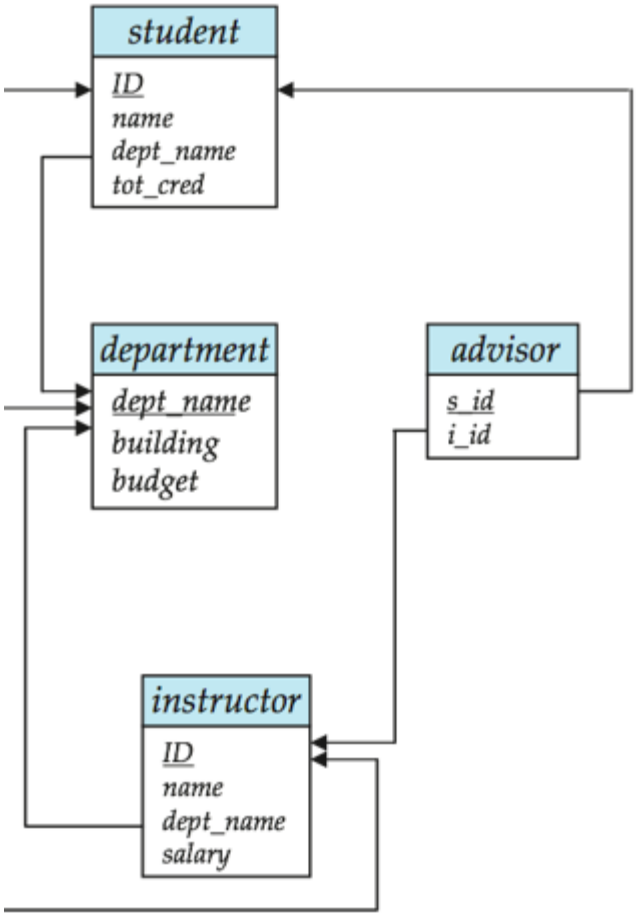


Discussion 3-11

- Represent the following query in SQL.

A. Total salary of instructors by department:
<dept_name, tot_salary>

B. Total salary of instructors by department, but only for departments whose total salary exceeds 2,000,000:
<dept_name, tot_salary>



Discussion 3-12

Show that, in SQL, **<> all** is identical to **not in**.

```
select *
from r1
where r1.a <> all (select a from r2)

select *
from r1
where r1.a not in (select a from r2)
```

Discussion 3-13

Consider the following SQL query. Does this query always correctly find values of *p.a1* that are either in *r1* or in *r2*? Discuss.

select distinct *p.a1*
from *p, r1, r2*
where *p.a1=r1.a1* **or** *p.a1=r2.a1*