CS 33007 Introduction to Database System Design, Fall 2017

Midterm Solutions

---

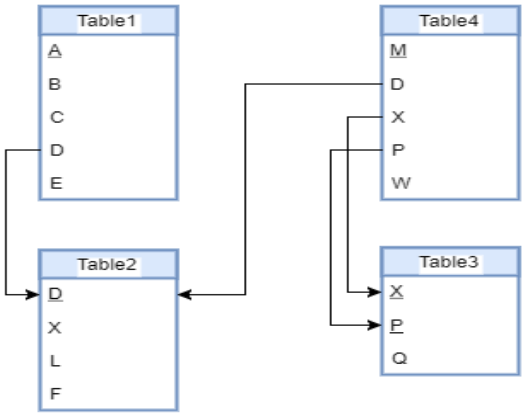
1. Consider the following database schema where primary keys are underlined.
   Table1 (A, B, C, D, E),
   Table2(D, X, L, F)
   Table3(X, P, Q)
   Table4(M, D, X, P, W)

   (i) Draw the schema diagram showing foreign key constraints. *[13 points]*
   (ii) Write a SQL query to create **Table1** considering datatype **varchar(20)** for all attributes.

   [12 points]

Answer

(i)



Note: We can't put foreign key constraints from X in Table2 to X in Table3 because X in Table3 alone can't form primary key for Table3.

(ii)
```
create table Table1(
        A varchar(20),
        B varchar(20),
        C varchar(20),
        D varchar(20),
        E varchar(20),
        primary key(A),
        foreign key(D) references Table2(D)
);
```

2. Answer following questions:

(i) Explain relationship between **super key**, **candidate key** and **primary key**. *[9 points]*

(ii) Write an equivalent SQL query using **natural join** for the following expression of relational algebra *[8 points]*

$$\Pi_{A, r.B, C, r.D, E}(\sigma_{r.B = s.B \wedge r.D = s.D}(r \times s))$$

(iii) Considering the following database Schema, *[8 points]*

classroom(<u>*building*</u>, <u>*room_number*</u>, *capacity*)
department(<u>*dept_name*</u>, *building*, *budget*)
course(<u>*course_id*</u>, *title*, *dept_name*, *credits*)
instructor(<u>*ID*</u>, *name*, *dept_name*, *salary*)
section(<u>*course_id*</u>, <u>*sec_id*</u>, <u>*semester*</u>, <u>*year*</u>, *building*, *room_number*, *time_slot_id*)
teaches(<u>*ID*</u>, <u>*course_id*</u>, <u>*sec_id*</u>, <u>*semester*</u>, <u>*year*</u>)
student(<u>*ID*</u>, *name*, *dept_name*, *tot_cred*)
takes(<u>*ID*</u>, <u>*course_id*</u>, <u>*sec_id*</u>, <u>*semester*</u>, <u>*year*</u>, *grade*)
advisor(<u>*s_ID*</u>, *i_ID*)
time_slot(<u>*time_slot_id*</u>, <u>*day*</u>, <u>*start_time*</u>, *end_time*)
prereq(<u>*course_id*</u>, <u>*prereq_id*</u>)

write an expression in relational algebra for following sentence,

Find the titles of courses in the Comp. Sci. department that have 3 credits.

Answer:

(i)

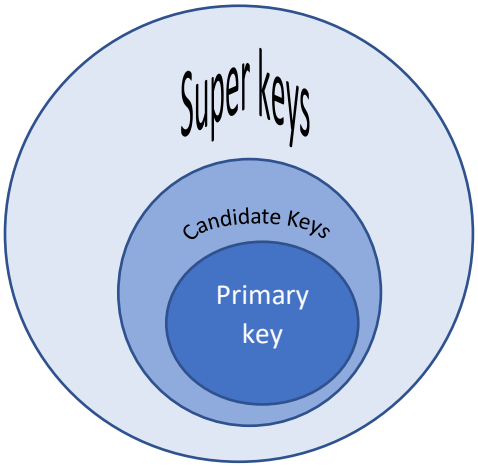


Fig. Relationship among super key, candidate key, and primary key. (Definitions in Chapter 2)

(ii) ***select*** *A, B, C, D, E* ***from*** *r* ***natural join*** *s;* *--- or ---* ***select*** *A, r.B, C, r.D, E* ***from*** *r* ***natural join*** *s;*

(iii) $\Pi_{title}(\sigma_{\text{dept name = 'Comp. Sci'} \wedge \text{credits=3}}(\text{course}))$

3.

(i) In a university, information about the instructors is stored in a relation called ***instructor(ID, name, dept_name, salary)***. It stores data about full professors, associate professors, assistant professors etc. The full professors get salary based on their years of service but at least $90,000. Write a SQL query to find names of full professors from Biology department. *[12 points]*

(ii) Considering schema in 2(iii), create a view which will allow corresponding users to see only name of the department with total salary of instructors of corresponding department. *[13 points]*

Answer

(i) **select** name **from** instructor **where** dept_name='Biology' **and** salary>=90000;

Note: In real scenario, salary scale for full professors is always higher than the scale for assistant and associate professors. You have to consider/assume this to answer this question.

(ii) **CREATE VIEW** dept_salary_view as **SELECT** dept_name, **sum**(salary) **from** instructor **GROUP BY** dept_name;

4.

(i) Considering database schema in 2(iii), define a function that, given the name of a building, returns the count of the number departments in that building. *[15 points]*

(ii) What is trigger? Explain with example, the purpose of using trigger in database.

[10 points]

Answer

(i)

```
create function dept_count (building_name varchar(20))
        returns integer
        begin
        declare d_count  integer;
            select count (* ) into d_count
            from department
            where department.building = building_name;
        return d_count;
    end
```

(ii) **Trigger:** A **trigger** is a statement that is executed automatically by the system as a side effect of a modification to the database.

The purpose of using trigger is to enable automatic operations on database based on the specific events. This helps to improve data integrity. Example can be anything supporting the definition above. It can be descriptive or using SQL

Ex1: In bank, some customers allow bank overdraft for the cost of daily charge on the overdraft amount. when a customer wants to withdraw money using check, a trigger can be used to check if the balance will go under a required minimum balance for the account and if it does, the trigger performs automatic actions on database such as sets a flag in database for sending notification to banker or account holder of the check, calculates and sets daily charge on overdraft amount etc.

More examples with SQL query can be found in chapter 5 (slides -26,27,28,29).