

# CS2102: Database Systems

## Tutorial #1: Relational Model & Relational Algebra

### Week 3

AY 2022/23 Sem 2

## 1 Discussions

The following questions are to be discussed during tutorial. All answers will be released with explanation.

1. **(Superkey)** Consider the following relation instance  $r$  of the relational schema  $R(A, B, C, D)$ .

R			
A	B	C	D
0	0	0	1
2	1	2	0
1	1	2	0
0	0	1	2

- (a) Assuming that  $r$  is a *valid* relation instance of  $R$ , write down all the *possible* superkeys of  $R$ .
  - (b) Additionally, suppose that it is also known that  $\{A, C\}$  is *definitely* a superkey of  $R$ . Based on the additional information, write down all the *possible* candidate keys of  $R$ .
  - (c) Which of these (*if any*) is likely to be the candidate key of  $R$ ?
2. **(Foreign Key)** Consider a relational database consisting of two relations with schema  $R(A, B)$  and  $S(W, X, Y, Z)$  such that  $A$  is the primary key of  $R$  and  $W$  is the primary key of  $S$  (*for future use, we denote this with  $R(\underline{A}, B)$  and  $S(\underline{W}, X, Y, Z)$  where the attributes being underlined are parts of primary key*).

Let  $r$  and  $s$  be the current instances of  $R$  and  $S$ , respectively, as shown below.

R		S			
A	B	W	X	Y	Z
3	0	0	4	0	NULL
2	1	1	NULL	2	NULL
1	1	2	1	2	NULL
0	0	3	0	1	NULL

Based on the current database instance above, write down all the *possible* foreign keys in  $S$  that refer to attribute  $A$  in  $R$ .

3. **(Equivalent)** Recap that two queries  $Q_1$  and  $Q_2$  on a relational database with schema  $R$  are defined to be **(strongly) equivalent** (denoted by  $Q_1 \equiv Q_2$ ) if for *every* valid instance  $r$  of  $R$ , either:

- both  $Q_1$  and  $Q_2$  queries produces error, *or*
- both  $Q_1$  and  $Q_2$  always compute the same results on  $r$

Consider a database with the following relational schema:  $R(\underline{A}, C)$ ,  $S(\underline{A}, D)$ , and  $T(\underline{X}, Y)$ , with primary key attributes underlined. Assume all the attributes have integer domain. For each of the following pairs of queries  $Q_1$  and  $Q_2$ , state whether or not  $Q_1 \equiv Q_2$ .

	$Q_1$	$Q_2$
(i)	$\pi_{[A]}(\sigma_{[A < 10]}(R))$	$\sigma_{[A < 10]}(\pi_{[A]}(R))$
(ii)	$\pi_{[A]}(\sigma_{[C < 10]}(R))$	$\sigma_{[C < 10]}(\pi_{[A]}(R))$
(iii)	$\pi_{[D, Y]}(S \times T)$	$\pi_{[D]}(S) \times \pi_{[Y]}(T)$
(iv)	$\pi_{[D, Y]}(S \times T)$	$\pi_{[D, Y]}(T \times S)$

4. **(Algebra)** Consider the following schema:

Relation	Description
$Pizzas(\underline{pizza})$	All the pizzas of interest.
$Customers(\underline{cname}, area)$	The name and location of each customer.
$Restaurants(\underline{rname}, area)$	The name and location of each restaurant.
$Recipes(\underline{pizza}, ingredients)$	The ingredients used in each pizza.
$Sells(\underline{rname}, \underline{pizza}, price)$	Pizzas sold by restaurants and the prices.
$Likes(\underline{cname}, \underline{pizza})$	Pizzas that customers like.

Additionally, we have the following foreign key constraints on the database schema:

- $(Recipes.pizza) \rightsquigarrow (Pizzas.pizza)$
- $(Sells.rname) \rightsquigarrow (Restaurants.rname)$
- $(Sells.pizza) \rightsquigarrow (Pizzas.pizza)$
- $(Likes.cname) \rightsquigarrow (Customers.cname)$
- $(Likes.pizza) \rightsquigarrow (Pizzas.pizza)$

Answer each of the following queries using relational algebra. For those interested, you can try writing the relational algebra query on <https://www.comp.nus.edu.sg/~cs2102/Tools/RelAlgebra/>. The syntax can be found in <https://www.comp.nus.edu.sg/~cs2102/Tools/RelAlgebra/help.html>.

You may find the sample data on the file T01.tbl.

- (i) Find all pizzas that Moe likes but is not liked by Lisa<sup>1</sup>.
- (ii) Find all customer-restaurant pairs  $(C, R)$  where  $C$  and  $R$  both located in the same area and  $C$  likes some pizza that is sold by  $R$ .
- (iii) Suppose the relation *Likes* contains all information about all customers. In other words, if the pair  $(cname, pizza)$  is not in the relation *Likes*, it means that the customer *cname* dislikes the pizza *pizza*. Write a relational algebra expression to find for all customers, the pizza that they dislike. The result should be of the form  $(cname, pizza)$ .

## 2 Challenge

The answers to the following questions is given without explanation. Please discuss them on Canvas.

1. **(Equivalent)** Recap that two queries  $Q_1$  and  $Q_2$  on a relational database with schema  $R$  are defined to be **(strongly) equivalent** (denoted by  $Q_1 \equiv Q_2$ ) if for *every* valid instance  $r$  of  $R$ , either:
  - both  $Q_1$  and  $Q_2$  queries produces error, *or*
  - both  $Q_1$  and  $Q_2$  always compute the same results on  $r$

Consider a database with the following relational schema:  $R(\underline{A}, C)$ ,  $S(\underline{A}, D)$ , and  $T(\underline{X}, Y)$ , with primary key attributes underlined. Assume all the attributes have integer domain. For each of the following pairs of queries  $Q_1$  and  $Q_2$ , state whether or not  $Q_1 \equiv Q_2$ .

	$Q_1$	$Q_2$
(i)	$(R \times \pi_{[D]}(S)) \times T$	$R \times (\pi_{[D]}(S) \times T)$
(ii)	$\pi_{[A]}(R \cup S)$	$\pi_{[A]}(R) \cup \pi_{[A]}(S)$
(iii)	$\pi_{[A]}(R - S)$	$\pi_{[A]}(R) - \pi_{[A]}(S)$

2. **(Algebra)** Consider the following schema:

Relation	Description
<i>Pizzas</i> ( <u>pizza</u> )	All the pizzas of interest.
<i>Customers</i> ( <u>cname</u> , area)	The name and location of each customer.
<i>Restaurants</i> ( <u>rname</u> , area)	The name and location of each restaurant.
<i>Recipes</i> ( <u>pizza</u> , ingredients)	The ingredients used in each pizza.
<i>Sells</i> ( <u>rname</u> , <u>pizza</u> , price)	Pizzas sold by restaurants and the prices.
<i>Likes</i> ( <u>cname</u> , <u>pizza</u> )	Pizzas that customers like.

Additionally, we have the following foreign key constraints on the database schema:

---

<sup>1</sup>The intention is to state it as "all pizzas that Moe likes but Lisa does not like", however, this may indicate dislike which we have not discussed the underlying assumption yet.

- $(Recipes.pizza) \rightsquigarrow (Pizzas.pizza)$
- $(Sells.rname) \rightsquigarrow (Restaurants.rname)$
- $(Sells.pizza) \rightsquigarrow (Pizzas.pizza)$
- $(Likes.cname) \rightsquigarrow (Customers.cname)$
- $(Likes.pizza) \rightsquigarrow (Pizzas.pizza)$

Answer each of the following queries using relational algebra. For those interested, you can try writing the relational algebra query on <https://www.comp.nus.edu.sg/~cs2102/Tools/RelAlgebra/>. The syntax can be found in <https://www.comp.nus.edu.sg/~cs2102/Tools/RelAlgebra/help.html>.

You may find the sample data on the file T01.tbl.

- For each restaurant, find the price of the most expensive pizzas sold by that restaurant. Exclude restaurants that do not sell any pizza.
- Find all customer-pizza pairs  $(C, P)$  where the pizza  $P$  sold by some restaurant that is located in the same area as that of the customer  $C$ . Include customers whose associated set of pizzas is empty.

- (Understanding RA)** Consider the following relational algebra query expressed on the database schema in Question 4.

$$\begin{aligned}
R_1 &:= \pi_{[pizza]}(\sigma_{[cname='Maggie']}(Likes)) \\
R_2 &:= \pi_{[rname]}(Sells) \times R_1 \\
R_3 &:= \pi_{[rname]}(R_2 - \pi_{[rname, pizza]}(Sells)) \\
R_4 &:= \pi_{[rname]}(Sells) - R_3 \\
R_5 &:= \pi_{[pizza]}(\sigma_{[cname='Ralph']}(Likes)) \\
R_6 &:= \pi_{[rname]}(\sigma_{[pizza5=pizza]}((Sells \times \rho_{[pizza5 \leftarrow pizza]}(R_5)))) \\
R_7 &:= R_4 - R_6
\end{aligned}$$

For each of the relational algebra expression  $R_i$ , write down a concise English sentence to precisely describe the information retrieved by  $R_i$ .