

## COMS 363 Fall 2022

## Assignment I

Percentage in your final grade: 9%

## Objectives:

1. Practice SQL Data Manipulation Language (DML) on a small database.
2. Practice recognizing advantages and disadvantages of different relational database designs

**Important:** The assignment is an individual assignment. You are welcome to consult the teaching staff, but you cannot share or reveal your code to any other students or looking up solutions from a previous class. Provide a citation to outside resources where you get an idea to do this assignment from.

The instruction for posting a private message on Canvas is available at

## Submission requirements

Submit A zip file named <netid>HW1.zip with the following files where <netid> is replaced with your university netid.

<netid>Q1.sql – add a comment into this file about which solution is for which problem.

<netid>Q2.pdf

The beginning of each file needs to include a comment indicating your authorship of the work, for example,

--Author: Wallapak Tavanapong

## Question 1. (75 points) Querying a small database.

Execute HW1\_Q1.sql in MySQL to generate the relational schemas and instances of the relations for the database about who attended which conference venue on which date at which location.

**venues** (venue\_id int, name varchar(200) not null unique, description varchar(500), **primary key(venue\_id)**)

**persons** (pid int, pname varchar(100) not null unique, **primary key(pid)**)

**events** (vid int, sched\_date date, location varchar(50), **primary key(vid, sched\_date)**, **foreign key(vid) references venues(venue\_id) on delete cascade on update cascade**)

**participates** (pid int, vid int, sched\_date date, **primary key (pid, vid, sched\_date)**, **foreign key(pid) references persons(pid)**, **foreign key(vid, sched\_date) references events(vid, sched\_date)**)

Write SQL queries to answer the following questions. Make sure your queries do not output any extra attributes or miss the required attributes. The output provided is for the instances of the relations in Fig. 1.

pid	pname	venue_id	name	description
102	Dean	1	Grace Hopper Conference	Conference for Celebration of ...
105	James	2	Science Bound	ISU program to empower Iowa...
103	Jane	3	Go Further	Program for Women in Enginee...
100	John	4	Computer Cyence	Computer science for high scho...
101	Mary	5	Iowa 4H Summer Workshop	Iowa 4H Youth Conference
104	Pak	6	Computational Thinking Compe...	K-12 outreach program offered ...
Persons instance		Venues instance		

vid	sched_date	location	pid	vid	sched_date
1	2017-10-20	Houston, TX	103	1	2017-10-20
2	2020-02-19	ISU Campus	101	2	2020-02-19
4	2021-08-02	Virtual	102	2	2020-02-19
4	2022-08-05	Principal head quarter, Des Moines	104	4	2021-08-02
6	2019-04-21	ISU Campus	104	4	2022-08-05
Events instance			100	6	2019-04-21
			101	6	2019-04-21
			103	6	2019-04-21
			Participates instance		

Fig. 1: Instance of the HW\_Q1 database

Your queries must give correct answers with the correct output column heading for all instances of this database.

- a. (5 points) Find the number of distinct events.

**Output:**

num_events
5

**Hint:** Use an appropriate aggregate function. There is no need to use distinct in your SQL query.

- b. (6 points) List all event(s) that were not held on ISU Campus. Show only the vid and sched\_date values of these events. You can use 'ISU Campus' in your query without worrying about the upper case or lower case.

**Output:**

vid	sched_date
1	2017-10-20
4	2021-08-02
4	2022-08-05

**Hint:** Use an appropriate condition for the where clause.

- c. (10 points) Find all venues and events that Pak participated in. Show the event vid, sched\_date, and location. Do not list duplicate rows. Use 'Pak' directly in your query. You do not need to use any wildcard, upper case, or lower case when specifying the names.

**Output:**

vid	sched_date	location
4	2021-08-02	Virtual
4	2022-08-05	Principal head quarter, Des Moines

**Hint:** Use three tables joining on the primary key and the foreign key(s) between the tables. No need to use distinct.

- d. (10 points) Find the event(s) that Pak or John have participated in. List all the attributes of the events ordered by the sched\_date value from the most recent event first. You do not need to use any wildcard, upper case, or lower case when specifying the names.

**Output:**

vid	sched_date	location
4	2022-08-05	Principal head quarter, Des Moines
4	2021-08-02	Virtual
6	2019-04-21	ISU Campus

**Hint:** Use order by in your query and a logical operator in your “where” clause.

- e. **(8 points)** Find the venue id that all three people, John, Mary, and Jane have participated in the events associated with that venue id. You do not need to use any wildcard, upper case, or lower case when specifying the names of the people.

**Output:**

vid
6

- f. **(7 points)** Find the name of the person(s) who has not participated in any event. Order the result in the descending order of the pid values. Use the right join operator in your SQL.

**Output:**

pname
James

- g. **(9 points)** Find the pid and pname of the person who attended at least two different events (different vid and sched\_date values).

**Output:**

pid	pname
101	Mary
103	Jane
104	Pak

**Hint:** Use a group by clause and a having clause to put an appropriate condition for the desired group properties.

- h. **(10 points)** Write a query with a sub-query in the where clause using the exists operator to find all the venues with no associated events. Show the venue\_id and name of the venue(s).

**Output:**

venue_id	name
3	Go Further
5	Iowa 4H Summer Workshop

- i. **(9 points)** Write a query with a sub-query in the where clause using the in operator to find the same information as in the question h.

**Output:**

Same as h.

- j. (1 point) Add your full name in the comment as the sole author of the submitted solution.

**Question 2: (25 points) Check your understanding of Relational Data Model**

Suppose you are given the following schema and an instance of the relation in Fig. 2.

**food\_ingredient**(fid int, fname varchar(45), iid int, iname varchar(45), amount int, caloriepergram float, category varchar(20), **primary key**(fid,iid))

Suppose your client is an owner of a restaurant. She tells you that the following functional dependencies hold.

$\{fid\} \rightarrow \{fname\}$

$\{iid\} \rightarrow \{iname, caloriepergram, category\}$

$\{fid, iid\} \rightarrow \{amount\}$

**Functional dependency definition:** In a relation schema R, a set of attributes X of R functionally determines a set of attributes Y of R ( $X \rightarrow Y$ ) if whenever any two tuples of R have the same value(s) for X, then they must have the same value(s) for Y.

For instance,  $\{fid\} \rightarrow \{fname\}$  means that for any two rows with the same fid value, those rows must have the same fname value.

Use the given relational schema and the relation instance to answer the following questions.

fid	fname	iid	iname	amount	caloriepergram	category
26	BBQ Chicken	23	Chicken	250	1.65	Meat
25	BBQ Pork	21	Pork	300	1.44	Meat
27	Fried Noodle	23	Chicken	200	1.65	Meat
27	Fried Noodle	28	Green Onion	60	0.1	Vegetable
27	Fried Noodle	29	Egg Noodle	250	2	Carbohydrate
18	Fried Rice	21	Pork	10	1.44	Meat
18	Fried Rice	23	Chicken	10	1.65	Meat
18	Fried Rice	28	Green Onion	10	0.1	Vegetable
30	Mongolian Beef	28	Green Onion	50	0.1	Vegetable
30	Mongolian Beef	31	Beef	300	1.8	Meat
22	Orange Chicken	23	Chicken	250	1.65	Meat
22	Orange Chicken	24	Orange	100	0.47	Vegetable

Figure 2. Instance of food\_ingredient

- a. **(5 points)** Given the above instance and the relational schema, suppose the restaurant owner wants to insert a new food item with fid of 28 and fname of “Mango and Sticky Rice,” but no ingredients are associated with this new food. With this schema design, is the owner able to insert the new food item? **Provide a reason to support your answer.**

**Answer (Yes/No):**

**Reason:**

- b. Another software developer gives alternate schema designs below. Answer the following questions. Primary keys are underlined.

**(5 points) Design 1**

**food** (fid int, fname varchar(45)); The primary key of food is fid.

**ingredient**(iid int, iname varchar(45), amount int, caloriepergram float, category varchar(20), fid int)

The primary key of the ingredient is iid; fid is a foreign key to the fid of the food table.

Can you store all instances of the food\_ingredient schema in the database in this design? If your answer is no, give an example of the row from Figure 2 that cannot be stored in the database with Design 1. If your answer is yes, give a reason why. The reason should not be specific to the rows in Figure 2.

**Answer (Yes/No):**

**Reason:**

**(5 points) Design 2:**

**food** (fid int, fname varchar(45))

The primary key of food is fid.

**ingredient**(iid int, iname varchar(45), caloriepergram float, category varchar(20))

The primary key of the ingredient is iid.

**recipe**(fid int, iid int, amount)

The primary key of recipe is only fid and the fid is a foreign key to fid in the food table.

Can you store all instances of the food\_ingredient schema in the database with this design? If your answer is no, give an example of the row from Figure 2 that cannot be stored in the database with Design 2. If your answer is yes, give a reason why. The reason should not be specific to the rows in Figure 2.

**Answer (Yes/No):**

**Reason:**

**(5 points) Design 3**

**food** (fid int, fname varchar(45), amount int, iid int); The primary key of food is fid; iid is a foreign key to the iid attribute of the ingredient table.

**ingredient**(iid int, iname varchar(45), caloriepergram float, category varchar(20));

The primary key of the ingredient is iid.

Can you store all instances of the food\_ingredient schema in the database with this design? If your answer is no, give an example of the row from Figure 2 that cannot be stored in the database with Design 3. If your answer is yes, give a reason why. The reason should not be specific to the rows in Figure 2.

**Answer (Yes/No):**

**Reason:**

**(5 points) Design 4:**

**food** (fid int, fname varchar(45))

The primary key of food is fid.

**ingredient**(iid int, iname varchar(45), caloriepergram float, category varchar(20))

The primary key of the ingredient is iid.

**recipe**(fid int, iid int, amount)

The primary key of recipe has two attributes: fid and iid. fid is a foreign key to fid in the food table; iid is a foreign key to iid in the ingredient table.

Can you store all instances of the food\_ingredient schema in the database in Design 4? If your answer is no, give an example of the row from Figure 2 that cannot be stored in the database with Design 4. If your answer is yes, give a reason why. The reason should not be specific to the rows in Figure 2.

**Answer (Yes/No):**

**Reason:**