**Homework Assignment 3**

Due: 11:59PM May 6, 2021

**1. (3 pt. each) Please answer to the following question in your own words.**

(a) (Exercise 6.5) An E-R diagram can be viewed as a graph. What do the following mean in terms of the structure of an enterprise schema?

- The graph is disconnected.

Ans) if the graph is disconnected, it means that each is not related.

- The graph has a cycle.

Ans)

when the graph is a cycle, it means that each entity set is connected, and all entity sets are related.

(b) (Exercise 6.17) Explain the difference between a weak and a strong entity set.

Ans)

unlike strong entities, weak entities do not have PK, but have discriminators, and all participate. However, a strong entity has a PK, and it may be not all participate.

(c) (Exercise 6.19) We can convert any weak entity set to a strong entity set by simply adding appropriate attributes. Why, then, do we have weak entity sets?

Ans)

weak entity has no PK. So, if we just add PK to weak entity from strong entity set. Then, that entity set become strong entity set.

(d) (Exercise 7.10) Our discussion of lossless decomposition implicitly assumed that attributes on the left-hand side of a functional dependency cannot take on null values. What could go wrong on decomposition, if this property is violated?

Ans)

if there is a null value on the left, there may be a case where the PK is null when decomposition is performed.

(e) (Exercise 7.21) Repetition of information and inability to represent information can be defined as next:

- Repetition of information: a condition in a relational database where the values of one attribute are determined by the values of another attribute in the same relation, and both values are repeated throughout the relation.

- Inability to represent information: a condition where there is a relationship that exists among only a proper subset of the attributes in a relation.

Explain why each of these properties may indicate a bad relational – database design.

Ans)

(1) If the information is repeated, then update abnormal, delete abnormal, insert abnormal and waste of storage are happened.

(2) When creating a new table through an entity set, there is no need for an entity set with inability to represent information, resulting in wasted memory.

(f) (Exercise 7.22) Why are certain functional dependencies called trivial functional dependencies?

Ans)

If any subset is dependent on the original set, it is trivial FD.

**2. Draw the E-R diagrams for the following databases. Be sure to indicate the cardinalities of the relationship.**

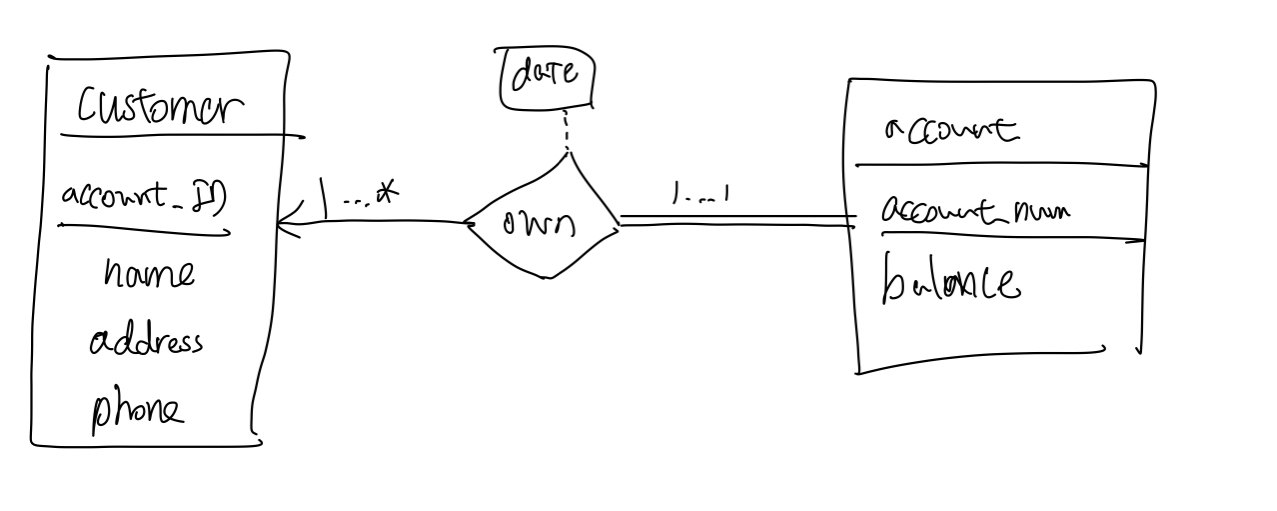
(a) (3 pt.) Design a database for a bank, including information about customers, their accounts, and the own relationship between them. Information about a customer includes their name, address, phone, and customer ID. An Account has an account number and balance. Also, the own relationship keeps opening date of each account. Note that:

- A customer can own multiple accounts.

- An account is owned by only one customer.

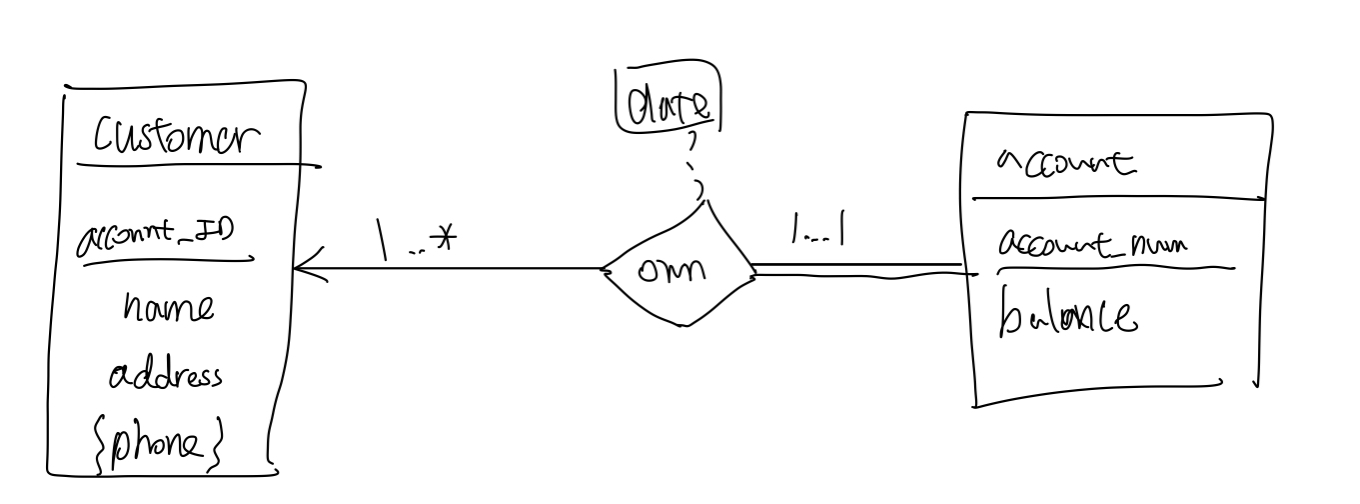
- Customer ID and Account number are unique to each customer and account, respectively.

Ans)



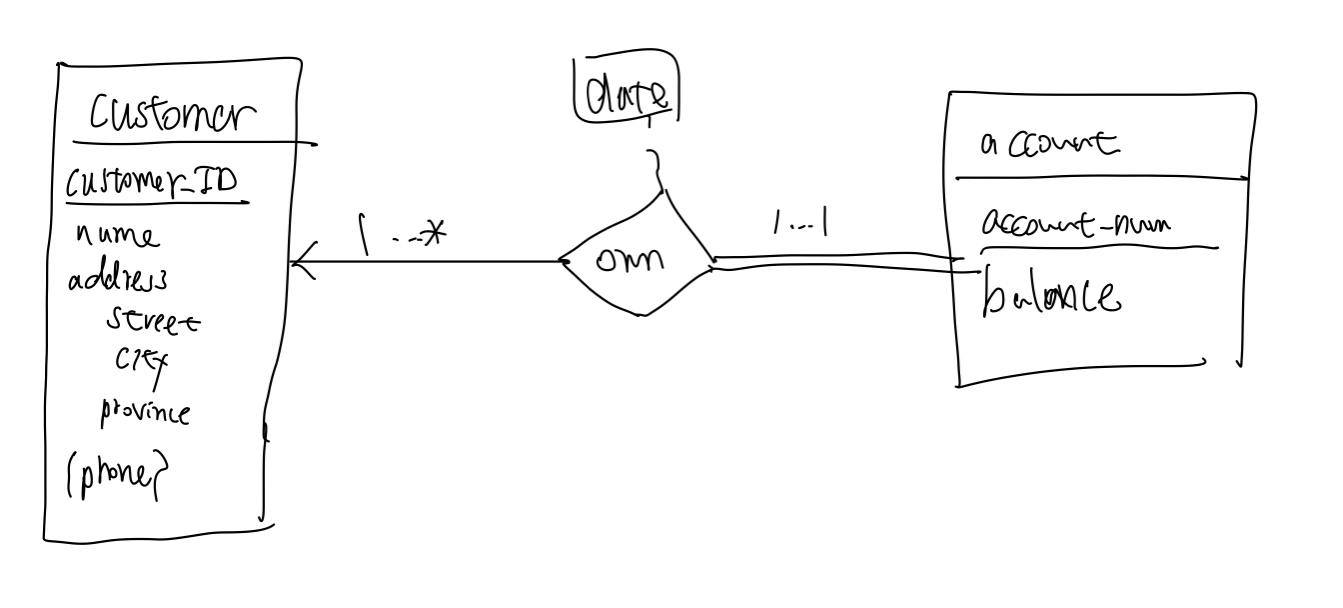
(b) (1 pt.) Modify your original diagram of Problem 2(a) such that a customer can have multiple phone numbers.

Ans)



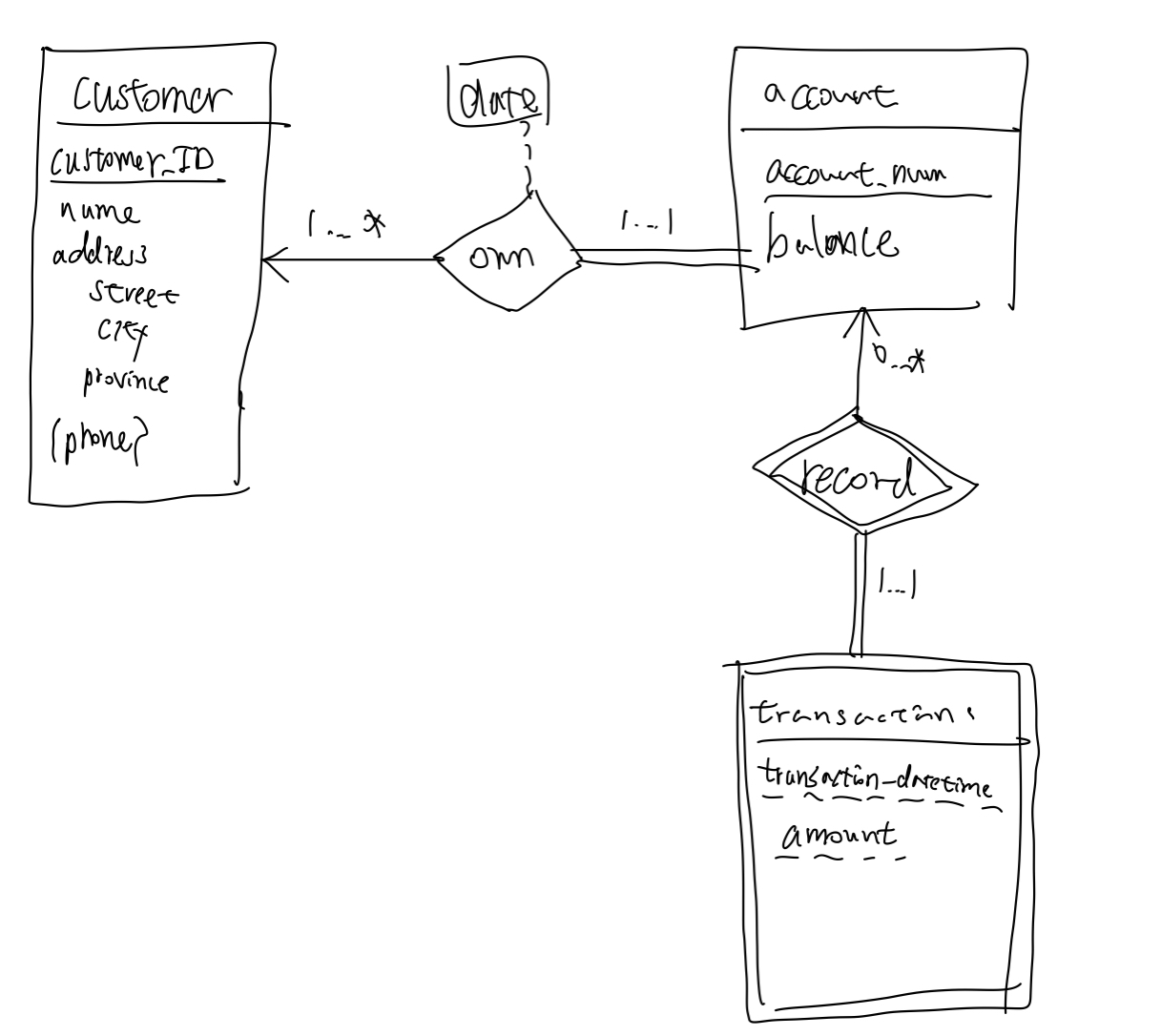
(c) (1 pt.) Change your diagram of Problem 2(b) such that a customer has an address represented by composite attributes (which are street-city-province triplets). Note that multiple customers may live at a single address.

Ans)



(d) (2 pt.) Add a weak entity set of transactions next to account (connected via a relationship record). This entity set contains transaction datetime and amount as its attributes.

Ans)



(e) (3 pt.) Convert the following E-R diagram into SQL DDL (CREATE TABLE statements).

- Consider the cardinalities of the relationship.

- Recall that we do not allow attributes to have non-primitive data types.

Ans)

CREATE TABLE customer (

customer\_ID INTEGER PRIMARY KEY,

name VARCHAR (30),

street VARCHAR (30),

city VARCHAR (30),

province VARCHAR (30)

)

CREATE TABLE phone (

customer\_ID INTEGER,

phone\_num VARCHAR (30),

PRIMARY KEY (customer\_ID, phone­\_num),

FOREIGN KEY (customer\_ID) REFERENCES (customer)

)

CREATE TABLE account (

account\_num VARCHAR (30) PRIMARY KEY,

balance VARCHAR (30),

date TIMESTAMP (14)

)

CREATE TABLE transaction (

account\_num VARCHAR (30),

transaction\_datetime TIMESTAMP (14),

amount INTEGER,

PRIMARY KEY (account\_num, transaction\_datetime, amount),

FOREIGN KEY (account\_num) REFERENCES (account)

)

**3. Normalization.**

(a) (3 pt.) Is every relation in 3NF also in BCNF? If yes, explain why. If no, given a counter example.

Ans)

No. when normal column can decide candidate key.

(b) (3 pt.) Is every relation in 4NF also in BCNF? If yes, explain why. If no, given a counter example.

Ans)

Yes. Because when complete BCNF, then, can do next step that is 4NF

(c) (3 pt.) The following relation violates {1NF, 2NF, 3NF, 4NF, BCNF}?

Ans)

2NF

(d) (3 pt.) The following relation violates {1NF, 2NF, 3NF, 4NF, BCNF}?

Ans)

2NF

(e) (3 pt.) The following relation violates {1NF, 2NF, 3NF, 4NF, BCNF}?

Ans)

3NF

**4. (4 pt. each) Join operations.**

(a) (Exercise 4.1) Consider the following SQL query that seeks to find a list of titles of all courses taught in Spring 2017 along with the name of the instructor.

SELECT name, title

FROM instructor NATURAL JOIN teaches NATURAL JOIN section NATURAL JOIN course

WHERE semester = ‘Spring’ AND year = 2017;

What is wrong with this query?

Ans)

Only instructor who taught in his department comes out.

(b) (Exercise 4.16) Write an SQL query using the university schema to find the ID of each student who has never taken a course at the university. Do this using no subqueries and no set operations (use an outer join).

Ans)

select ID  
from student left outer join takes using(ID)  
group by ID  
having count(course\_id) = 0;



(c) (Exercise 4.17) Express the following query in SQL using no subqueries and no set operations.

SELECT ID

FROM student EXCEPT

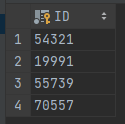
SELECT s\_id

FROM advisor

WHERE i\_ID IS NOT NULL;

Ans)

select ID  
from student s left outer join advisor a on s.ID = a.s\_ID  
where i\_ID is null;



(d) (Exercise 4.20) Show how to define a view tot\_credits(year, num\_credits), giving the total number of credits taken in each year.

Ans)

CREATE VIEW tot\_credits (year, num\_credits) AS (

SELECT year, SUM (credits)

FROM students

GROUP BY year

)

(e) (Exercise 4.21) For the view that you have defined in the previous problem (Problem 4(d)), explain why the database system would not allow a tuple to be inserted into the database through this view.

Ans)

There is a GROUP BY clause, so when I do insert, then the rejection is occurred.

‘

**5. SQL query time!**

Launch and access the MySQL databases distributed with the class virtual machine. Below uses the “sakila” database (DVD rental database), which consists of 16 tables regarding movie inventory, actors, customers, rental history, payment information, etc. For each of the following questions, find the answer based on the information recorded in the database and write a query that shows how you obtained the answer.

(a) (2 pt.) How many stores are found in the database?

Answer to the question:



Query to find the answer:

select count(store\_id)  
from store;

(b) (2 pt.) How many unique last names are found in the actor relation?

Answer to the question:



Query to find the answer:

select count(distinct last\_name)  
from actor;

(c) (2 pt.) According to the database, how many inventories (DVDs) have not been returned (inventories that have not been returned do not have return\_date)?

Answer to the question:



Query to find the answer:

select count(rental\_id)  
from rental  
where return\_date is null;

(d) (2 pt.) How many distinct customers have rented a movie title(s) from staff\_id=1?

Answer to the question:



Query to find the answer:.

select count(distinct(customer\_id)) as count  
from customer join rental using(customer\_id) join staff using(staff\_id)  
where staff\_id = 1;

(e) (3 pt.) How many distinct films rated 'PG' are available?

Answer to the question:



Query to find the answer:

select count(distinct(title)) as count  
from film  
where rating like 'PG';

f.

(f) (3 pt.) How many active customers are living in the district of England?

Answer to the question:



Query to find the answer:

select count(customer\_id)  
from customer join address using(address\_id)  
where district = 'England' and active = 1;

(g) (4 pt.) Considering the rental history (rental) and payment history (payment), who has paid the largest amount of money for renting movies? List the first and last name of the customer, the total number of movie rentals, and total amount of money s/he has paid.

Answer to the question:

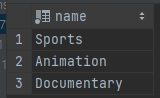


Query to find the answer:

with payment as (  
 select customer\_id, sum(amount) as sum  
 from payment  
 group by customer\_id  
), rental as (  
 select customer\_id, count(rental\_id) as count  
 from rental  
 group by customer\_id  
)  
select first\_name, last\_name, count, sum  
from customer join payment using(customer\_id) join rental using(customer\_id)  
where sum >= all(select sum from payment);

(h) (4 pt.) List three most common categories of film available at store\_id=2 (if a store has multiple copies of the same film, consider each copy as an individual inventory). Tip: Use LIMIT 3 at the end of your query to limit the number of output tuples.

Answer to the question:



Query to find the answer:

with category\_count as(  
 select category\_id, count(film\_id) as count  
 from film\_category join inventory using(film\_id)  
 where store\_id = 2  
 group by category\_id  
)  
select name  
from category\_count ct, category c  
where ct.category\_id = c.category\_id  
order by count DESC limit 3;

(i) (3 pt.) What is the title of the movie that has the longest description (film\_text.description) among the rental store with store\_id=2 has?

Answer to the question:

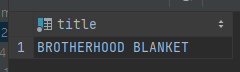


Query to find the answer:

select distinct title  
from film join inventory using(film\_id)  
where store\_id = 2 and length(description) >= all(select length(description) from film);

(j) (4 pt.) Which of the films starred by "FRED COSTNER" rented the most? Write the title of the film.

Answer to the question:

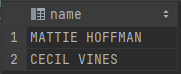


Query to find the answer:

with temp as (  
 select film\_id, count(film\_id) as count  
 from rental  
 join inventory using (inventory\_id)  
 join film\_actor using (film\_id)  
 join actor using (actor\_id)  
 where first\_name = 'FRED' AND last\_name = 'COSTNER'  
 group by film\_id  
)  
select f.title  
from temp as t, film as f  
where t.count = (select max(count) from temp) and t.film\_id = f.film\_id;

(k) (2 pt.) Using the 'customer\_list' view, list all names of people whose address is in the city of 'London'.

Answer to the question:



Query to find the answer:

select name  
from customer\_list  
where city LIKE 'London';

(l) (3 pt.) Write a query that uses only tables (does not use any views) and returns the same information as in the previous problem (Problem (k)).

Answer to the question:

select concat(first\_name, ' ',last\_name) as name  
from customer join address using(address\_id) join city using(city\_id)  
where city like 'London';

Query to find the answer:

