#Created exam with SQL and EER diagram
#Original work (Q+A)

1. Logic

   Does (¬(P ∧ ¬Q) ∧ Q) |= ((P → Q) ∧ Q)? Prove or Disprove, showing all work.

Answer:

P	Q	¬P	¬Q	P∧¬Q	¬(P ∧ ¬Q)	(¬(P ∧ ¬Q) ∧ Q)	(P → Q)	((P → Q) ∧ Q)	A→B
0	0	1	1	0	1	0	1	0	1
0	1	1	0	0	1	1	1	1	1
1	0	0	1	1	0	0	0	0	1
1	1	0	0	0	1	1	1	1	1

True, A → B

2. EER Modeling

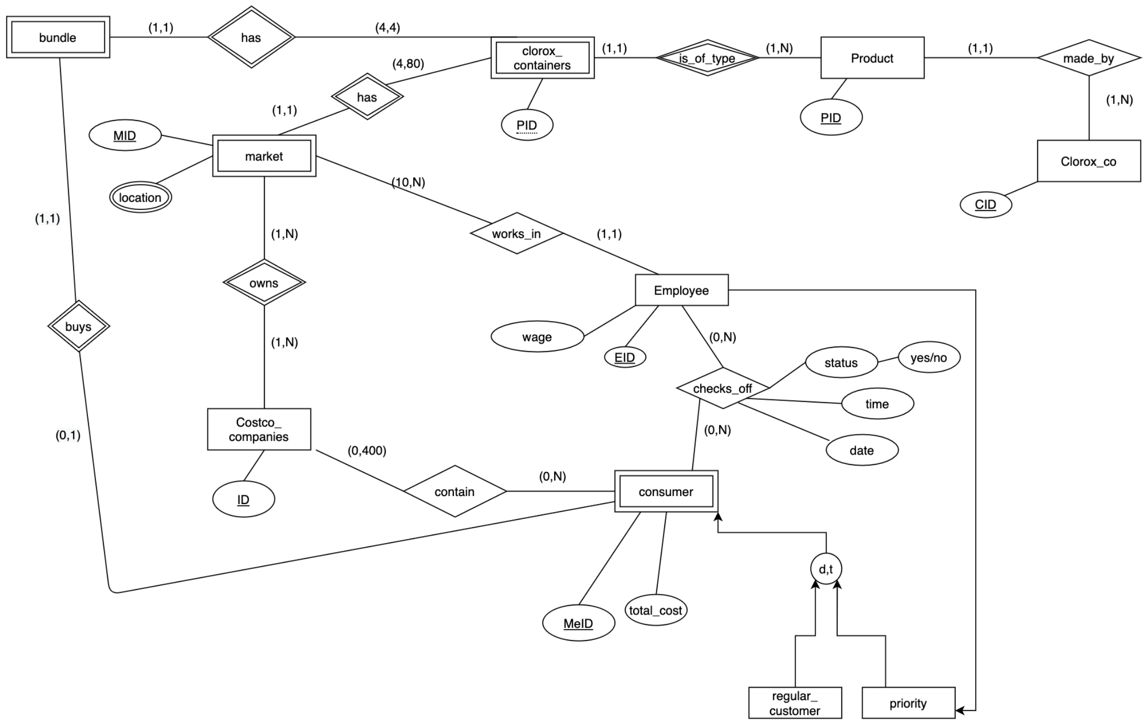
In response to COVID-19, California's government wants to assess how successful supermarkets are in distributing Clorox wipes. You are hired to organize the following information into a database.

- For every supermarket (or market), we record the market's store ID, the company that owns it (Costco Companies), and the location. Each supermarket has 80 and less available wipes for everyone each month (we are only doing a database for 1 month) depending on the supply. We can assume that the supermarket will at least have one bundle each month. The products will have a unique ID since it is a product made by the Clorox Company.
- For each customer, we record the total cost of their purchases (only regarding the wipes), and their member ID. No customer can buy more than 1 bundle of wipes each month, which is equivalent to 4 clorox containers. We can also assume that each supermarket can hold up to 400 consumers at a time.
- For each employee that works in the market, they will have an employee ID that is unique. In each market, you can assume that the employees have the same rules as regular consumers. There are at least 10 workers available in the store depending on how many customers there are. An employee will record the customers who have bought clorox wipes and their names. Just like how Costco checks-off each consumer by the exit, the employees will check-off each consumer as they leave the store and record them by member ID alone with the time and date if they have bought clorox wipes. You can assume that employees are consumers too.

Draw an EER Diagram with *exactly 6 Regular Entity Types (including sub- classes), 4 Weak Entity Types, and 10 Relationship Types (including weak-entity relationships)*. Clearly label details, including primary and partial keys and (max,min) constraints. List any additional assumptions you make.

Please do your work on the next page.

This page is available for your EER Diagram.


bundle
(1,1)
has
(4,4)
clorox_
containers
(1,1)
is_of_type
(1,N)
Product
(1,1)
made_by
(1,N)
Clorox_co
CID
PID
PID
(4,80)
has
(1,1)
MID
market
location
(10,N)
works_in
(1,1)
(1,1)
(1,N)
owns
(1,N)
Employee
wage
EID
(0,N)
checks_off
status
yes/no
time
date
(0,N)
buys
(0,1)
Costco_
companies
ID
(0,400)
contain
(0,N)
consumer
MeID
total_cost
d,t
regular_
customer
priority

3. Queries in Algebra and SQL

Consider a TikTok user's database for their sales on Etsy. In the relations below, we store the user's information, their activity, and the success they have made since their first appearance on TikTok to advertise their small business.

1. seller_user (SID, lname, fname, MI, phone_number)
2. activity (SID_1, interactions, month, year, day)
3. item (IID, name, benefits, cost, quantity, month, year, day)
4. success (SID_1, IID_3, gain, loss, revenue)

For each query below, write both (ALG) relational algebra expressions and (SQL) statements. If it is impossible to formulate a query in one of the languages explain why. Clearly list all assumptions you make.

(a) How many interactions did the seller have in 2016 when they gained more than and equal to 50 dollars?

(ALG)

Answer:

R ← π interactions (σ gain >= 50 AND year = 2016 (activity * success))

(SQL)

Answer:

SELECT interactions
FROM activity AS a, success AS s
WHERE a.SID = s.SID AND s.gain >= 50 AND a.year = 2016;

(b) What is the difference in cost average for the rose quartz item from August and July of 2016?

(ALG)

Answer:

R1 ← π cost, name, month (item)

R2 ← month $\mathcal{F}$ <AVG cost> (R1)

R3 ← π cost (σ name = 'rose quartz' AND month = 'August' AND year = 2016 (R2))

R4 ← π cost (σ name = 'rose quartz' AND month = 'July' AND year = 2016 (R2))

R ← R3 - R4

(SQL)

Answer:

(SELECT AVG(a.cost) FROM item AS a WHERE a.month = 'August' AND a.year = 2016) -
(SELECT AVG(b.cost) FROM item AS b WHERE b.month = 'July' AND b.year = 2016) AS difference

(c) At what quantity does revenue equal cost?

(ALG)

Answer:

$R1 \leftarrow \pi\ quantity, revenue, cost\ (item * success)$

$R2(revenue, quantity) \leftarrow \pi\ quantity, revenue\ (R1)$

$R3(cost, quantity) \leftarrow \pi\ quantity, cost\ (R1)$

$R \leftarrow \sigma\ R2 < R3\ (R2 \cap R3)$

(SQL)

Answer:

```
SELECT quantity
FROM item AS i, success AS s
WHERE i.IID = s.IID AND i.cost = s.revenue;
```

(Prefered answer):

```
SELECT quantity
FROM item AS i INNER JOIN success AS s
ON i.IID = s.IID AND i.cost = s.revenue;
```

(d) Which item (by name) has the largest range total (regard gain and loss) overall in 2016? Assume loss to be a positive number.

(ALG)

Answer:

$R1 \leftarrow \pi\ name, gain, loss\ (item * success)$

$R2(IID1, gain) \leftarrow \pi\ gain\ (\sigma\ year = 2016\ (R1))$

$R3 \leftarrow year\ \mathcal{F}\ <MAX\ gain>\ (R2)$

$R4(IID2, loss) \leftarrow \pi\ loss\ (\sigma\ year = 2016\ (R1))$

$R5 \leftarrow year\ \mathcal{F}\ <MAX\ loss>\ (R4)$

$R \leftarrow R3 - R5$

(SQL)

Answer:

```
SELECT name
FROM item AS i
WHERE year = 2016 AND EXISTS (SELECT MAX(gain) AS mgain, MAX(loss) AS mloss,
                                (mgain - mloss)
                                FROM success s
                                WHERE  i.IID = s.IID);
```

#largest range total (use max for both since largest loss and largest gain)

4. Short Answer.

(a) Give two examples of axioms that were found from Godel's Incompleteness Theorem. Give an example of a topic in our class where we see Godel's Incompleteness Theorem and briefly explain why.

Answer:

If x equals y and y equals z, then x must equal z. Another example is if a line was drawn to connect two points.

This is significant in our class because Godel's Incompleteness Theorem explains that all true statements don't always have a proof and that is clearly shown when we are working on formal logic (i.e. truth tables versus proposition).

(Note: because numbers, truth tables aren't words, it makes less sense and you realize that there is a gap between what seems to be true and the proof of it)

(b) When converting subclasses and superclasses to relations, which constraints work for programming-major and mathematics-major in a student superclass? Explain why. What could the relations look like?

Answer:

We can use "o" (overlapping), "d" (disjoint), "p" (parital), and "t" (total) because students can be working towards either a programming major or a mathematics major, the student can also be in one or the other major, some students can also be in neither, and lastly, all students can be either programming majors or mathematic majors. Therefore, all cardinalities are compatible with this relation.

1) Student (<u>SID</u>, grade)
   1a) programming_major (<u>SID1</u>, coding)
   1b) mathematics_major (<u>SID2</u>, mathematics)

(c) Name 1 superclass (entity) from your Team Design Project and label the cardinality constraint accordingly with the subclasses.

Answer: Contact (superclass); Customer, Shipper, and Manufacturer (subclass)

Cardinality = overlapping and partial (o,p)