Homework 7

due March 30, 2025, via D2L

Problem 1:

Create the following database in MySQL using appropriate constraints such as foreign key restraints.

orders(ord_no, client_no, date_received, date_shipped)

orderDetails(ord_no, item_no, quantity, price)

client(client_no, clientName, address)

item(item_no, itemName, description)

Obviously, an order needs to have not only an ord_no, but also an existing client. Constraints are: date_received cannot be null, date_shipped as a default value of '9999-01-01' that will get updated when the order is shipped. Similarly, an orderDetails item needs to have an existing item in the table item. Notice that both ord_no and item_no in orderDetails need to be the primary key. I messed this up in a previous version.

(a) Populate the database with four clients, five items, twenty orders with on average two items ordered.

As we have seen, MySQL does not have materialized views. However, we can create a table and fill it with data.

(b) Create a materialized view, implemented as a MySQL table that shows the client name, the item name, the date the order was received for all orders that have not been shipped.

(c) Instead of a materialized view, we create a table that contains the name and the address of all clients who have gold standard, meaning that they ordered at least x dollars worth of items. You should pick x commensurate with the price of the items you invent.

Write triggers that update this table whenever an order is created (i.e. an insert into orders), updated (i.e. an update in orders or in orderDetails), or canceled (i.e. a delete into orders).

Run test runs to ascertain that this table is maintained correctly as clients receive or loose gold standard.

Deliverables: All the code that you needed to run this example. The contents of the tables from (a), (b), and the table defined in (c). Workbench screen captures are fine.