Term Project

Goal: The goal of this project is to provide a realistic experience in the conceptual design, logical design, implementation, operation, and maintenance of a small relational database.

Application description: The application is an electronics vendor that operates both a Web site and a chain of many physical stores. To find out more about this application, think about any experiences you may have had making purchases both online and in-store, and browse their Web sites.

Here are a few points to consider:

* There are many different products, grouped into a variety of (possibly overlapping) categories. Groupings can by type of product (cameras, phones, etc.), by manufacturer (Acer, Apple, etc.), or by other means (for example, an ASUS PC might be packaged with a Sony monitor and an HP printer and marketed as a package).
* Some customers have a contract with the company and bill their purchases to an account number. They are billed monthly. Other customers are infrequent customers and pay with a credit or debit card. Card information may be stored for online customers, but not for in-store customers.
* Online sales must be sent via a shipper. The company needs to store the tracking number for the shipping company so it can respond to customer inquiries.
* Inventory must be accurate both in stores and in warehouses used to replenish stores and to ship to online customers. When inventory is low, a reorder should be sent to the manufacturer and listed in the database. When goods arrive, inventory should be updated and reorders marked as having been filled.
* Sales data are important for corporate planning. Marketers may want to look at sales data by time period, product, product grouping, season, region (for stores), etc.

Project requirements:

1. E-R Model

* Construct an E-R diagram representing the conceptual design of the database. You must include all the entity and relationship sets implied by this handout.
* Be sure to identify primary keys, relationship cardinalities, etc.

2. Relational Model

* After creating an initial relational design from your E-R design, refine it based on the principles of relational design we are studying in the course.
* Create the relations in a database of your choice.
* Create constraints as appropriate.
* If as you refine your design, you discover flaws in the E-R design, go back and change it. Your final E-R design must be consistent with your relational design.

3. Populate Relations

* Include enough data to make answers to your queries interesting and nontrivial.
* You may write a program to generate input data. You may share such programs and/or data with your classmates, but NOT your overall schema design.

4. Queries:

You should run a number of test queries to see that you have loaded your database in the way you intended. The queries listed below are those that your client (the manager from the vender company) wants. They may provide further hints about database design, so think about them at the outset of the project.

* Find the customer who has bought the most (by price) in the past year.
* Find the top 2 products by dollar-amount sold in the past year.
* Find those products that are out-of-stock at every store in Kaohsiung.
* Find those packages that were not delivered within the promised time.
* Assume the package shipped by USPS with tracking number 123456 is reported to have been destroyed in an accident. Find the contact information for the customer. Also, find the contents of that shipment and create a new shipment of replacement items.

There are several types of users who access the database. Each may need a special application

* Customer service needs a lookup application to check inventory both locally and at nearby stores.
* Call center staff need an application that allows quick access to customer data and the ability quickly to enter phone orders.
* The stocking clerks at the warehouses need an application to help them record incoming shipments and update inventory.

5. What to turn in:

Usually, I ﬁnd the ﬁrst checkpoint requires some discussion, while checkpoint 2 can often be handled quickly. Please talk to me about questions at any point; don’t wait for a check point. The ﬁnal version of the project is to be turned in on paper. Please take the time to put the items together in a readable format.

* E-R diagram, plus any explanatory notes (I will accept paper for the ER diagram)
* Relational schema
* Do NOT turn in a printout of all your data. Just run select count(\*) on each of your tables so I can see how big they are without having to go online.
* The code you wrote for each of the listed queries (in GitHub, GitLab or other platforms) and the result from running the queries.
* A README file in the top-level folder that explains what is where, etc. Include usage instructions for the interfaces