SA8902 Database Management and Spatial Technologies

Assignment #1: Relational Database Construction

Objectives

This assignment is designed to help you understand the following:

1) How to perform data normalization for effective information management;

2) How to implement a simple relational database;

3) How to ask questions of the database using saved queries/sql.

Description

"Bracer Inc." is a new online business that sells products tailored to the personal fitness industry such as exercise machines, free weights, and exercise clothing with their main warehouse based in Toronto. Currently, all their records are managed and maintained using a simple spreadsheet to keep track of their orders [see spreadsheet: BracerInc.xls]. However, the business is expanding and the spreadsheet is becoming unmanageable. You have been hired to design and construct a SQL Server database that will help Bracer Inc. to effectively keep track of their information.

As part of your contract, you are required to provide a detailed explanation of the various steps involved in constructing the database. The specific components they require documentation for are as follows:

1. Describe the normalization process used to separate the BracerInc.xls spreadsheet into the various tables you include in the database. What specific steps were required to get from the original spreadsheet to the database design that you created? What was the justification for each separate table and how were the columns chosen for each table identified?
2. Describe the modification problems that are currently present in the BracerInc.xls spreadsheet and how your database design solves these problems.
3. Provide an E-R Diagram for your database that implements the Crow's Foot notation. This can be constructed using any kind of graphic design or data modeling software you wish.
4. Provide a detailed explanation of your E-R Diagram that explains the Entities in your database and the relationships between them. This should include an explanation of the minimum and maximum cardinality rules your E-R model implements.
5. A data dictionary that provides a technical description of each table in your database.
6. A set of SQL Server query files (saved using the given filenames) that answer the following requirements:

    a. Show a list of customers sorted by name and showing just their names, city and postal codes. (Save as Q1).

b. Show a list of customers who spent less than $200. Give their names and the amount they spent sorted in descending order of amount spent. (Save as Q2).

c. Show the names of the customers who gave a rating of 4 and the product name to which the customers gave a rating of 4. (Save as Q3).

d. Show the names of the customers who live outside Toronto. Give their names, postal codes and the products they bought. (Save as Q4).

e. Show a list of incomplete orders. Give the order ID, customer name., product ID, product name, and shipping date. (Save as Q5).

f. Show the products out of stock. Give the product name, product category, and cost. (Save as Q6).

g. Show exactly what each customer from London has ordered. The information should include customer ID, customer name, description of each item, quantity and price of each item. (Save as Q7)

Database Creation Steps

1. Normalize the data structure represented in the BracerInc.xls spreadsheet. Take notes on this process as you create each table for your description to answer question 1.
2. After normalizing the data and preparing your ER diagram, use SQL Server Management Studio to create a new database.
3. Design the tables that will hold the normalized data.
4. Set up the **primary key** for each table
5. Specify any **constraints** for each attribute in a table.
6. Once each of the relations (or tables) is specified, link primary key and foreign key between the tables.
7. Finally populate the tables in your database with the normalized data (see step 1) in each table.
8. Create and save the seven query files.

Deliverables

1. A final report, both hard copy and saved as a Microsoft word document.

2. A SQL Server database backup file containing your completed database. Backup your database following the instructions in Lab2 and save the backup file as **LastName-FirstName-Assignment1.bak**, but replace this filename with your own first and last name. If using the Remote Desktop then generate a .sql script of your database and submit that .sql file instead of a .bak file.

3. Create and save the seven SQL Server query files (.sql) to answer the questions listed. Include hard copies of these queries as an appendix at the end of the report.

Submission

Digital versions of the final report (.doc, .docx), final SQL Server database backup file (.bak) and the SQL Server query files (.sql) should be submitted on D2L no later than **4:00pm on the due date specified in the course outline**. In addition to these digital files, **a hard copy print out of the report and .sql files should be handed in at the beginning of class**.