Project Assignment

General. Your project is to design and implement a database for a supermarket inventory and check-out system. At the back-end, the database manages all of its data in the Oracle database management system. At the front-end, you are required to implement a command-line interface to its users using Java and JDBC (although if you know how to build a web interface, you can do so for extra credit).

I will expect certain minimal functionality in your database - beyond that, the sky's the limit. Minimal functionality includes a variety of queries and browsing capabilities over the current inventory in the store (e.g. the user should be able to search by product name, description, etc), the transactions, and customers. The user should be able to update the database by inserting or deleting tuples, e.g. imagine the scenario when a customer checks out – the purchased products are automatically entered into the database by scanning the barcodes. Of course, in this project, you won't need to worry about this layer of interface (the barcode scanner). But your command-line interface should allow transaction info to be entered into the database manually.

The followings are the “basic” information that the database should store:

1. Stock information: product name, product description (e.g. 12 16-oz bottles of Coke), category (e.g. Fruit, Vegetable, Meat, Beverage, etc.), brand, marked price, Universal Product Code (UPC), quantity

2. Supplier and Supplies information: name, address, phone number, products (note a product can have more than one supplier)

3. Transaction information: transaction ID, transaction date, products purchased, quantities, customer ID.

4. Sales information: products, sales prices, promotion start date, promotion end date

5. Customer information: customer ID, customer name, age, gender

After each transaction, customers should know how much money they have saved, if any.

After each transaction, the quantities of (affected) products in the inventory should be updated automatically.
Each product is classified into different categories such as Fruit, Vegetable, Meat, Pet, etc. For simplicity, assume that each product belongs to only one category.

When users search for an product by name or description, the search should retrieve all products whose names or descriptions contain the search terms, e.g. if the search term is “Coca Cola” on product name, then all products whose names contain “Coca Cola” should be retrieved.

Note: The descriptions above summarize the minimal information your database should contain. It does not necessarily imply the organization of the relations. You may re-arrange the attributes or add more attributes. The design decision is yours, but your design should be reasonably efficient, and well justified.

You’ll be graded for each phase. The project accounts for 20% of your total grade.

**Phase 1: Conceptual Design.** (4%) You need to submit a corresponding ER diagram according to the project description and the assumption you have made. (See HW1 assignment for an example).

**Phase 2: Schema Design.** (4%) After your conceptual design is finished, you need to translate the ER diagram into relation schema. Submit a copy of your relation schema, the SQL script that you use to implement your database, and insert enough tuples (e.g. 50 products, 10 suppliers, 20 customers, 20 different transactions – one for each customer. Each customer can buy multiple items in one transaction) into each table. Also submit a copy of your original (with the TA’s feedback) and the revised ER designs. This is required if you want to get full credit for Phase 2.

**Phase 3: JDBC Implementation.** (10%) Write a program in JDBC that will allow users to access and query your database. Your program should allow users to do the following:

- Add records: Enter information for new customers, new products, new suppliers, and new transactions. Update/delete information.
- When entering a new transaction, display the amount saved for each product immediately after the product has been entered. At the end of the transaction, display the total amount saved for this transaction, and the total amount spent so far by this customer.
- Search database: search products based on product names, description, etc.
- Display all products that are low in stock (those that have under 100 in stock).
- Display all products that have not been purchased so far.
- Display some statistics: customer who spent the most money so far, transaction that has the largest number of (unique) products, the best-selling product (by count) so far.

(2%) Prepare a final report. Your report should contain materials from all phases. Discuss and justify your design decisions, and challenges you encountered.
**Timeline:**

Oct 25: project released

Nov 3: Phase 1 (ER diagram) due

Nov 8: Phase 1 returned

Nov 15: Phase 2 (schema) due

(Nov 29): HW4 due

Dec 8: Phase 3 and final report due