

## **Project** **Order Processing** **System**

You are required to analyze, design and implement a database system to support the operations of a simplified online bookstore. Also, you are required to create the necessary forms to perform these operations. The data and functional requirements of the system are as follows.

### **PART 1**

#### **I. Database relations**

For each book in the store, the system keeps the book's ISBN number, title, author(s), publisher, publication year, selling price, and category.

The book's category can be one of the following: "Science", "Art", "Religion", "History" and "Geography". The system has to keep track of the names of publishers, their addresses and telephone numbers.

Information about book orders is also maintained.

#### **II. Operations**

Assume that the user of the system can do the following operations: -

##### **1. Add new books**

To add a new book, the user enters the properties of the new book along with a threshold (the minimum quantity in stock to be maintained for that book).

##### **2. Modify existing books**

For updating an existing book, the user first searches for the book then he does the required update. For a given book, the user can update the quantity in stock when a copy or more of the book is sold. The user cannot update the quantity of a book if this update will cause the quantity of a book in stock to be negative. **(hint: trigger before update)**.

##### **3. Place orders on books**

An order with constant quantity is placed only when the quantity of a book drops from above a given threshold (the minimum quantity in stock) to below the given threshold **(hint: trigger after update in books table)**.

##### **4. Confirm orders**

The user can confirm an order when receiving the ordered quantity from the book's publisher; the quantity of the book in store automatically increases with the quantity specified in the order. Assume that deleting the order means that the order is received from publisher. **(hint: trigger before deletion from orders table)**.

##### **5. Search for books**

The user can search for a book by ISBN, and title. The user can search for books of a specific Category, author or publisher.

### **III. Physical Design**

Do an appropriate physical design assuming the following statistics on data volume and work load statistics. The statistics listed below are important for physical database design. However, they are used for **illustration purposes** and are not intended to characterize any specific real system.

- Number of books in the database: 1,000,000
- Average Number of authors: 500,000
- Number of inquiries to the database (using the book ISBN, title or category) per day: 150,000
- Number of customer purchases per day: 50,000
- Average number of book orders generated per day :10

### **PART 2**

In Part 1, you designed a database for the central management of a book store; in this session, you are required to update this database to be used for online management and procurement.

- Your main task is to discriminate between two types of users: managers and customers and add some functionalities to each type as will be detailed below.
- Only previously registered users can login to the system.
- New customers are able to sign up for a new customer account by providing the necessary information: user name, password, last name, first name, e-mail address, phone number, and shipping address.
- A registered customer can do the following activities:
  1. Edit his personal information including his password
  2. Search for books by any of the book's attributes. (Use indices to speed up searches when possible)
  3. Add books to a shopping cart
  4. Manage his shopping cart. This includes the following.
    - View the items in the cart
    - View the individual and total prices of the books in the cart
    - Remove items from the cart
  5. Checkout a shopping cart
    - The customer is then required to provide a credit card number and its expiry date. This transaction is completed successfully if the credit card information is appropriate.
    - The book's quantities in the store are updated according to this transaction.
  6. Logout of the system
    - Doing this will remove all the items in the current cart.

Managers can do the same operations that a normal customer can do; a manager has the following additional Options:

1. Add new books
2. Modify existing books
3. Place orders for books
4. Confirm orders
5. Promote registered customers to have managers credentials
6. View the following reports on sales
  - a. The total sales for books in the previous month
  - b. The top 5 customers who purchase the most purchase amount in descending order for the last three months
  - c. The top 10 selling books for the last three months

Assume that the system stores book sales and other related data for the last 3 months.

**Notes:**

- Preserve all the integrity constraints through relational model constraints or triggers and use Commit and Rollback commands at appropriate places.
- Use **Java** to create the required forms, and you can use **Jasper** library or any appropriate way for the reports.

**Deliverables:**

- You are required to provide the database generation script (SQL file), the source code for the java forms and any related codes (jasper .jspx files, etc ..).
- You are required to submit a report containing the following:
  - The database ERD diagram in a readable format (Simple MySQL generation needs to be adjusted in order to become user readable).
  - Detailed analysis of all the triggers, procedures, indices, any database tuning and concurrency / transaction controls measures done throughout the project. You need to provide justification for any choices you made.
  - Sample screenshots of your program demonstrating most of the functionality.
- Deadline for the project is on June 13.
- Discussions time will be decided at a later time.