Project 2: Amazon Competitor Database

In this project, you will design, implement, and test a database for an Amazon competitor/clone. In your design, you will need to determine

a) necessary information to store (using second page as *suggestions*)
b) an E/R diagram to construct (using MS Visio or any similar *tool*)
c) business reports to produce (using at least four *views*)
d) business logic to incorporate into the design (using several *triggers and constraints*)
e) security levels to establish (using *scripts* for password assignments, roles and encryptions)
f) performance and efficiency improvements to apply (using at least four *stored procedures* and four *functions*)
g) regular business transactions (using at least four *transactions*)

In your implementation, you will need to

a) determine the tables, columns, primary keys, datatypes, nullabilities, and relationships.
b) normalize your design into its 3^{rd} Normal Form.
c) address potential integrity and security issues.

In your testing, you will need to

a) populate your database with test data (about 10 rows in each table)
b) produce about five reports (please use also your views)
c) demonstrate its reliability through several complex scenarios
d) perform several transactions like customer purchasing goods from the company, company gets goods from the supplier, customers leave reviews for the supplier, customer perform returns, etc.

Please submit your typed report as a single file (.doc or .pdf) on Blackboard. Your report should include following sections:

1. [5] Cover page: a descriptive title, a short abstract, and your information
2. [50] Design: your introduction, design considerations and choices, and E/R diagrams
3. [10] Implementation: source codes with your comments, and screenshots of your design
4. [10] Testing: testcases and screenshots with your comments
5. [5] Conclusions: your project analysis, and your remarks on this project

All reports that satisfy all the following basic requirements will have a base score of 20 points. The maximum points deducted for noncompliance is indicated in parentheses.

1. General report guidelines followed, i.e. sections labeled, pages numbered [10]
2. Report adheres to basic standards of grammar and spelling [10]

SYRACUSE UNIVERSITY ENGINEERING & COMPUTER SCIENCE

Project 2: Project Requirements

In this project, you will design, implement, and test a database for an Amazon competitor. Detailed requirements are provided below in addition to the requirements in the original Project 2 statement. Please follow this list carefully as it will be used as part of the grading rubric.

Design:

1) Include everything provided in Project 2 statement file. Points will be deducted based on the percentage of missing pieces.
2) Only MS Visio or similar tool is allowed for your design. Do NOT use SQL server management studio to generate your design graph.
3) Please do NOT cross lines between tables and make your design easy to read. A full screenshot of your design is required, which should also be readable.
4) CamelCase naming standard is required (If a name is formed of multiple words that are joined together as a single word, the first letter of each of the multiple words should be capitalized and no underscore is allowed). Incorrect capitalization will lose points.
5) Please use full word to name your tables/columns. Do NOT use shortcuts.

Implement:

1) Determine primary keys, data types, nullabilities and relationships reasonably. These should follow the real world. For example, do NOT use “Table1” or “Person2” as your database entry content. Every record of data should have real meaning, like “PersonInformation” etc.
2) Please code the creation of database (including tables, data insertions, queries, basically every part that needs SQL code) by yourself. No auto-generation of SQL code is allowed.

Test:

1) For data loading to test your database, each table should have around 10 records with real meaning (same as the Requirement 1 in the above Implementation part). Proper datatypes are required, using meaningful data to demonstrate your database work properly.
2) For testing part, views/functions/stored procedures/scripts/transactions will be used to verify your database design. Again, scenarios should be meaningful. Any scenario like “Create a view to show all customers’ information” will NOT be accepted.
3) For every scenario, please add a short description to demonstrate the purpose of that scenario. Please use SELECT/EXEC to verify results of your scenarios.
4) To test your database, the design and data everyone loaded must be different. Teamwork is NOT allowed.

Business Problem:

Your job is to create a database for an Amazon competitor. This database is to contain the following suggested information but more shall be added as you deem necessary. Please note that each order has a single shipping address, and can contain multiple products from one or more warehouses. Multiple warehouses may have the same product and the originating warehouse for a shipment is chosen based on the shipping cost, which can be determined by the zip codes of the originating warehouse and the shipping address. Suppliers may also have multiple products in various warehouses.

Customers:

- Name
- Username
- Email
- Phone numbers (home, cell, business)
- Address book (several shipping and billing addresses)
  - City
  - State
  - Street
- Orders/Returns Information
- Wishlist
- Reviews
  - Date when the review is left
  - Which Order
  - Score
  - Text
  - Which product

Products:

- ProductID
- ProductName
- Description
- Price
- Customer ratings:
  - Text
  - Score
  - Date when created
  - Customer who left it

Suppliers:

- SupplierID
- Name
- Address
- Phone
- EmailAddress
- Products information:
  - ProductID
  - Number of Products Available

Orders/Returns:

- OrderNumber
- Status (Ready to go, Shipped, Delivered, Return)
- OrderDate
- OrderItems:
  - ProductID
  - ProductName
  - Quantity
  - UnitPrice
- TotalPrice
- Shipping Service (USPS, FedEx, …)
- Shipping Address
  - City
  - State
  - Street
- Shipping fare (depends on the address between the warehouse and the shipping address)
- Expected Shipping date
- Actual Shipping date
- Shipping Information

Warehouses:

- WarehouseID
- Address
  - City
  - State
  - Street
- Stored Products:
  - ProductID
  - SupplierID
  - ProductName
  - Number in stock
  - Number on the way
  - Number in return