UMBC

**IS 722: Systems & Information Integration**

Fall 2019

**Project Phase I**

**Due Date**: 11:59PM on Thu, Oct 17, 2019

In this first phase you are going to design and implement a simple database application. This phase is to be implemented by each student individually, without the help of others. You will design and implement one of the following databases (details for each one are in the next pages):

* Inventory database, or
* Ordering database

The choice of database is up to you. Feel free to pick one of them – they are equivalent in terms of work to be done.

**Generic instructions for all databases:** Your database should have at least 3 tables. Each table should have several columns with descriptive names and it should be populated with a small number of records (around 10 for each table). You may use primary/foreign keys for your tables.

You need to implement a set of operations for your assigned database in SQL or PL/SQL. More details on these operations are found in the detailed description of each database.

**Deliverables**: Upload on Bb a file (Word, or PDF) with the following:

1. [20 pts] ER Diagram
2. [20 pts] Script starting with delete commands (deleting all tables), then all create commands, and some insert commands per table
3. [60 pts] Operations: Implementation and execution of the 6 database operations as follows: For each operation include:
   1. The SQL (or PL/SQL) code implementing each operation
   2. Screen shots with the execution of the operations showing that your database application actually works

**Inventory Database**

Your inventory database contains information about books, and music CDs. Customers buy products and their purchases are recorded in the database. You should store information with details about all the products, customers, and their purchases.

Below is a set of mandatory fields for your database tables. You will probably need additional columns, and possibly tables, to make it functional and support the inventory operations.

* Books: ISBN (unique identifier), title, author(s), price, category (fiction, drama, etc.), keywords (to describe its contents), etc.
* Music CDs: ID (unique identifier), title, artist(s), recording company, price, genre (e.g. classical, jazz, pop, rock), etc.
* Customers: name, address, phone #, product, etc.
* Purchases: product sold, customer, quantity, amount, etc.

**Operations to be implemented for the inventory database**:

1. List all products in inventory (just a few columns for each product)
2. List all details of a product given its unique identifier
3. List each product(s) of a particular author or artist
4. List each product with at least quantity 4
5. List each product that is out of stock
6. List each product that is bought by at least 3 customers

**Ordering database**

Customers order goods and provide their credit card number for payment purposes. This database keeps track of the customer orders and billing information. It contains detailed information about each customer and each purchase order.

Below is a set of mandatory fields for your database tables. You will probably need additional columns and possibly tables to make it functional and support the ordering operations.

* Customers: Customer name, address, phone number, email
* Product: Product ID (unique identifier), name, description, quantity, price, etc.
* Orders: Order# (unique identifier), order date, customer, one or more products ordered, total price (includes product prices, sales tax, shipping)
* Line Items: Item# (unique identifier), description, quantity ordered, price per unit

**Operations to be implemented for the Ordering database**:

1. List all orders in the database (just a few columns for each order)
2. List all details of an order, given its unique identifier
3. List all orders of a particular customer
4. List each order with at least 3 different items
5. List all orders that have been ordered after a specific date of your choice
6. List the customer name and a total number of orders by that customer