

MEMO

DATE: February 11, 2021
TO: Cambria Art Gallery
FROM: Leonard Margaria

SUBJECT: Development tasks

DUE DATE: Monday February 22, 2021 11:59pm

SQL Chapters needed to complete this task:

SQL 1: Retrieving Data Using the SQL Select Statement
SQL 2: Restricting and Sorting Data
SQL 3: Using Single-Row Functions to Customize Output
SQL 4: Using Conversion Functions and Conditional Expressions (NVL function)
SQL 6: Displaying Data from Multiple Tables (JOIN, OUTER JOIN)
SQL 9: Manipulating Data (INSERT)
SQL 10: Using DDL Statements to Create and Manage Tables (CREATE TABLE)
SQL 11: Creating Other Schema Objects (CREATE VIEW)

Professor Floyd has shared with me some of the ERDs that have been created regarding the data used in the operations of my art gallery. I have a number of different teams developing this system for my business and to help me compare the systems developed and to assure quality, I am asking that you and your team do the following tasks.

- STATE ANY ASSUMPTIONS YOU HAVE MADE
- CHECK YOUR WORK TO ASSURE IT IS CORRECT
- IF YOU ARE UNABLE TO ACCOMPLISH ANY PART OF THIS TASK, THIS MUST BE STATED AT THE BEGINNING OF YOUR SUBMISSION. NO PARTIAL CREDIT WILL BE GIVEN FOR THIS TASK IF IT CAN BE EASILY DETERMINED THAT YOU HAVE SUBMITTED WORK THAT IS INCORRECT.
- CHECK YOUR SCRIPT TO ASSURE YOUR CODE MATCHES YOUR ERA DIAGRAM.

TASK 1: Customer and Preferences Tables

1. Creating Tables:
 - a. The attributes of a Customer can be seen on my Sales Invoice. All of these attributes should be required. Phone and email should be unique. Be sure to store first and last names separately; make the State attribute have a DEFAULT value of 'CA'.
 - b. For the Preferences table, include art type, artist, description, start_date and end_date. Assure end_date is after start_date. The art type and start date attributes are required. Make the start date have a default of the date a preference record (instance) is entered into the table.
Note that a preference is associated with only one customer and must be associated with a customer.
2. Data:
 - a. Enter 10 customer records into the Customer table.
 - i. 5 customer INSERTs must illustrate use of the DEFAULT clause and the rest must not make use of the DEFAULT clause

- b. Enter 8 preference records in the Preference table as follows
 - i. 1 customer instance (e.g., our customer Bob) will be associated with 3 preference instances (e.g., he wants a ceramic, a painting, and something else for example). This results in 3 INSERT statements putting data into the PREFERENCE table.
 - ii. 1 customer instance should be associated with 2 preference instances.
 - iii. 3 different customer instances should be associated with 1 preference each (that is there are 3 different customer and 3 different preferences and so 3 different associations).
 - iv. 5 customers should not have a preference (that is, there is record in the PREFERENCE table associated with these 5 customers).
 - v. 3 preference instance INSERTs must illustrate of the DEFAULT clause and the rest must not make use of the DEFAULT clause
3. Quality Control Check List ... assure that the following are functional. In some cases, this can be demonstrated by the data that is entered. In other cases, you can test them by entering data that tried to violate the constraint. You do not need to submit anything to show that you have performed these activities but your system should work correctly.
 - a. CHECK: End Date is after Start Date
 - b. UNIQUE: constraints work (phone, email)
 - c. NOT NULL:
 - d. Reference integrity: These are tasks you can do to explore the issue of referential integrity.
 - i. Changing the value of a customer's PK when that customer has a preference record.
 - ii. Deleting a customer when a customer has a preference.
 - iii. Deleting a preference.
 - iv. Changing a preference from one customer to another customer (e.g., say that a preference has been associated with customer 1000, and you want to change the FK to a different customer (e.g., customer 2000) where customer 2000 exists.
 - v. Changing a preference from one customer to another customer (e.g., say that a preference has been associated with customer 1000, and you want to change the FK to a different customer (e.g., customer 2000) where customer 2000 does not exist.
 - vi. Adding a preference to a customer who does not exist.
4. Queries: Write the following SELECT statements that returns the following to test your work. Write these as CREATE VIEW statements.
 - a. List of customers including the following attributes: first name, last name, street, city, state, zip, phone, email ordered by last name.
 - b. List of customers and their preferences. Only include customers who have preferences. Include the following attributes: first name, last name, phone, art type, artist, start date, end date.
 - c. List of customers and their preferences including customers who do not have preferences. Include the following attributes: first name, last name, phone, art type, artist, start date, end date. For those customer who do not have preferences change the value of art type to the phrase "No Preference".

Submission:

SQL SCRIPT consisting of the following:

Comments at beginning of script with the names of your team members.

DROP TABLE STATEMENTS (put all DROP TABLE statements at beginning of script)

CREATE TABLE

INSERT STATEMENTS (put insert statements after each table)

CREATE VIEW (put all CREATE VIEW statements at end of script)

NOTE: THIS SCRIPT FILE IS NOT TO BE SUBMITTED IN WORD BUT IN A FILE THAT CAN BE OPENED AND EASILY READ IN NOTEPAD. THE SCRIPT FILE SHOULD BE EXECUTABLE IF I UPLOAD IT TO ORACLE APEX.

An accompanying PDF document that presents any assumptions, things that don't work, etc.