Lab 4

Due Oct 6, 2020 by 11:59pm **Points** 10 **Submitting** a file upload
Available until Dec 16, 2020 at 11:59pm

This assignment was locked Dec 16, 2020 at 11:59pm.

Labs 4 to 7 are SQL scripts; if they do not run in their entirety without error, they will be graded 0.

Statement of Authorship

- Lab to be done individually
- In order to be graded, the following Statement of Authorship must be present at the beginning of the script
- I, Firstname Lastname, student number 123456789, certify that this material is my original work. No other person's work has been used without due acknowledgment and I have not made my work available to anyone else.
- Replace Firstname Lastname with your name and 123456789 with your student number

Introduction

Normally, an instance of SQL Server would reside on a central database server that would be accessible throughout an enterprise. Once databases and tables are created they would persist. However, the environment at the College is different. There is no central server for you to persist your database; therefore it must be recreated throughout the semester. In order to simplify this process, you will create a database script in this lab. You will expand this script in labs 6 and 7.

Overview

- Start SQL Server and SQL Server Management Studio, see **lab 0** for instructions on how to use SQL Server and SSMS
- Take an existing database creation script and modify it for your tables and columns
- Add an index creation to the script for an attribute in the master table
- Add a view creation to the script

Naming Conventions

- Do not put spaces in the names of anything you create
- Do not use descriptive prefixes or postfixes such as tbl
- Use all lowercase and divide word boundaries with _, known as snake_case
- See **http://www.sqlstyle.guide** **(http://www.sqlstyle.guide/#naming-conventions)**

Script Modification

Download the sample **co859.sql** script which can also be found on the Student Resources page. Edit the script; you may edit in Notepad or any text editor. However, it will be better to edit with SQL Server Management Studio, because this will provide colour syntax highlighting and syntax checking. Update the comment heading block to reflect your name as a modifier of the script, the date you modified it, and your business.

Script Start

The first part of the script is setting up the environment in which the remainder of the script will run. This includes setting the date format to ymd which will accommodate the sales record INSERTS that have their date format in YYYY-MM-DD. If you used a different date format in previous labs, switch to YYYY-MM-DD for this lab as it is the safest date format. Also, if the co859 database already exists, it is deleted. It is then created either for the first time or for any subsequent time. Having the conditional delete allows the script to run numerous times without having to manually delete anything. Do not rename the database. In a script, each SQL statement should be terminated with a semicolon. You will notice the SQL Server keyword GO throughout the script. This is an indication to the database that a batch of SQL statements is done and is ready to be processed by the DBMS. These are necessary any time the structure of the database changes.

SQL Keywords

SQL Server Management Studio uses colour syntax highlighting. Ensure that none of your identifiers (tables, columns, constraints, etc.) are SQL keywords or reserved words. If any of your identifiers change colour from black to any other colour, it is a reserved word and you must change it.

Master Table

Because Dr. Darla provides a service, the sample script creates a table named dental_services. Your master table name must be **customers**, **items** or **<description>_services** (don't worry if you previously named this table something else in labs 2 or 3), update the CREATE TABLE statement (and comment) appropriately. You will probably also have to update the column and constraint details to match your work from lab 2. Feel free to use different column names and data types than you used in labs 2 or 3.

Sales Table

Your sales table name must be **sales** (don't worry if you previously named this table something else in labs 2 or 3). You will probably also have to update the column and constraint details to match your work from lab 2. Feel free to use different column names and data types than you used in labs 2 or 3.

Loading the Master Table

The master table needs to be loaded with at least 5 records using INSERT statements. You may edit the sample INSERT statements in place or use the BuildInserts Excel workbook to generate the INSERT statements.

Loading the Sales Table

The sales table needs to be loaded with at least 15 records using INSERT statements. You may edit the sample INSERT statements in place or use the BuildInserts Excel workbook to generate the INSERT statements.

Creating an Index

SQL Server automatically creates an index for the primary key in a table. This feature means that you as a DBA don't have to worry about establishing indexes for this column. Nonetheless, you would still want some indexes set up. Typically, the indexes would be on columns commonly used for accessing records in the tables.

- If your master table is a customers table, create an index on customer name
- If your master table is an items table, create an index on item description
- If your master table is a services table, create an index on service description
- The naming convention for indexes is IX_table_name_column_name

Syntax for the CREATE INDEX statement:

```
CREATE [UNIQUE] INDEX index_name
ON table_name (column_name1 [ASC|DESC] [, column_name2 [ASC|DESC]]);
```

ple CREATE INDEX statement:

```
CREATE INDEX IX_patients_last_name
ON patients (last_name);
```

Note:

1. It is up to you to determine if your name or description column will have duplicates (they probably won't)
2. Square brackets indicate optional elements
3. Add this CREATE INDEX statement to the script

Creating A View

Create a view that highlights the most expensive elements of your master table. The creation of a view needs to be the first thing in a batch, so put a GO statement after the CREATE INDEX command to start a new batch. To produce this information, you will create a view named **high_end_<master>** (where <master> is customers, items or services) in your script as follows:

- 15 characters of name or description field (not the whole column), use the SUBSTRING function, when using SUBSTRING in a view, the column must have a column alias
- Year to date sales
- Only select records where the hourly rate or credit limit or quantity on hand (depending on which master table you have) is greater than average, you will need a nested subquery to achieve this

Sample CREATE VIEW statement:

```
CREATE VIEW current_patients
AS
SELECT patient_id, SUBSTRING(primary_diagnosis, 1, 25) AS diagnosis, nursing_unit_id
FROM admissions
WHERE discharge_date IS NULL
```

Verify Inserts

Update the insert verification code, if necessary. The insert verification should be the last thing in the script.

Lab 4

Criteria	Ratings			Pts
Script comment block updated appropriately	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
Master table (Customer, Item or ServiceMaster) CREATE TABLE statement	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
Sales CREATE TABLE statement	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
Master table INSERT statements	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
Sales table INSERT statements	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
CREATE INDEX statement	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
CREATE VIEW statement	**2 pts** **Full Marks**	**1 pts** **Partial Marks**	**0 pts** **No Marks**	2 pts
All database objects named correctly and no SQL keywords or reserved ds used for identifiers	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
Script comments present (including the INDEX and VIEW) and appropriate	**1 pts** **Full Marks**	**0.5 pts** **Partial Marks**	**0 pts** **No Marks**	1 pts
				Total Points: 10