Computer Science 1600
Assignment 3 (Distance Section)
Spring 2019
Due on Tuesday, July 16, 2019,
before the cut off time of 11:59pm (Newfoundland Time)

Associated Lab Work
Labs 6-7

Files to be used with this assignment: COMP1600_A3_S19.accdb, COMP1600_A3_S19_ DESIGN.docx

For this assignment, download the given file **COMP1600_A3_S19.accdb**. Open the database and study the following:

- the structure (Design View) of each of the tables included in the database. Study the fields, their data types and the description of the fields (specifically the primary and foreign keys, if any).
- the Relationship table (accessed from the **Database Tools** tab, **Relationships** option) that shows all the tables in the database and how they are related to each other.

Using this file, first **design** and then **implement** the queries listed below. Type your design of the queries into the given **MS-Word** document, called **COMP1600_A3_S19_ DESIGN.docx**, by completing the **query design table** and the **relationship design table** for each query.

Each row of the query design table corresponds to a row in the actual query design window for **MS-Access** (see Figure 1) as described below.

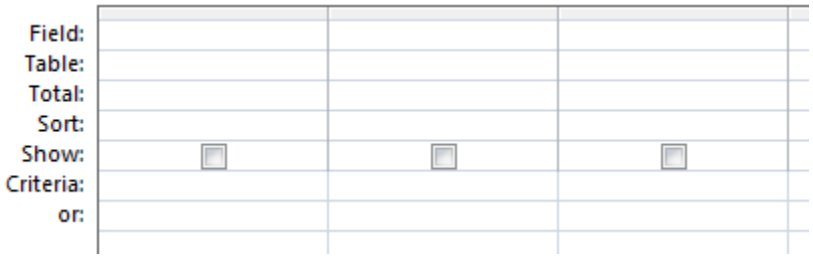


Figure 1

Query Design Table:

Field:										
Table:										
Total:										
Sort:										
Show(Y/N):										
Criteria:										
Or:										

- **Field:** used to list the name of the fields that are required to generate the desired results.
- **Table:** used to list, for each field, the name of the table from where each of the fields are taken.
- **Total:** used to identify whether the listed fields in the column should be grouped or have a built-in function applied by specifying **Group By** or the built-in function name (**SUM**, **AVG** etc.)

- **Sort:** used to list the order (Ascending or Descending) of the sort, for a sort field(s).
- **Show (Y/N):** used to indicate whether the field should be displayed or not (Y/N) (sometimes you do not display every field that you have listed in the "Field" row).
- **Criteria:** used, if needed, to list the criteria or condition required to select the records you want displayed based on the specific field(s).
- **Or:** used, if needed, to list alternative "or" criteria to select the records you want displayed based on the specific field(s).

Note: As with all rows (Table, Sort, etc), the condition(s) listed on the Criteria row are listed in the column, under the appropriate field name where it is to be applied. Additionally, note that the equal sign is optional for conditions checking for equality.

Relationship Design Table (Join Table):

From Table	From Field	To Table	To Field

- **From Table:** the name of a table (usually for one of the main entities)
- **From Field:** the field name of the primary key for that table
- **To Table:** the name of a related table
- **To Field:** the field name of the foreign key for that related table

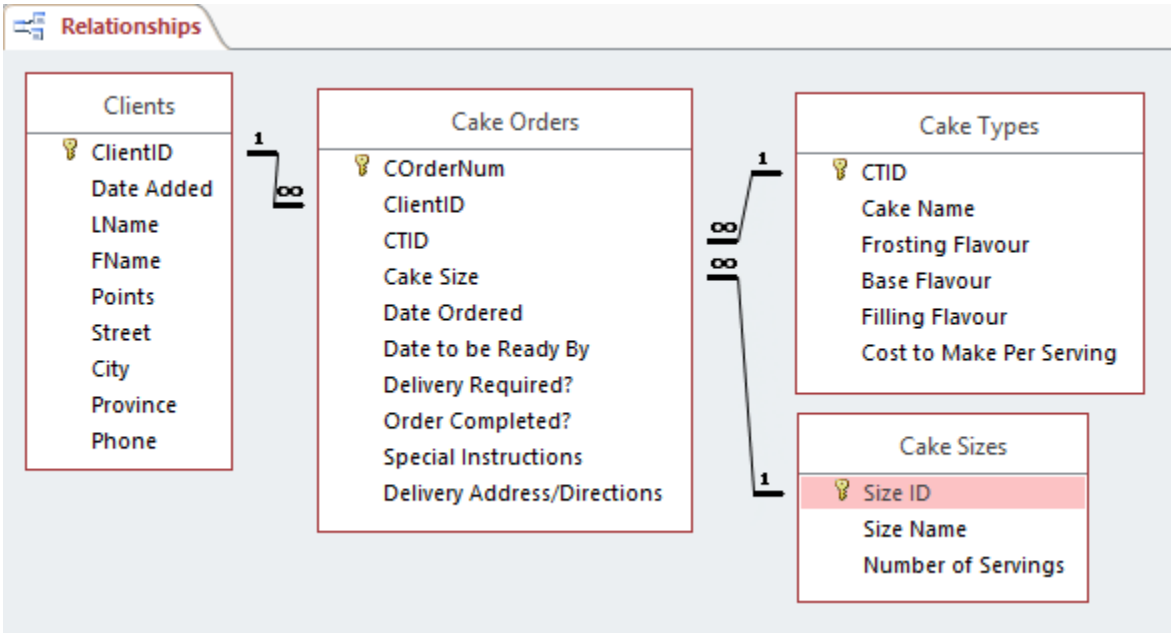
Note: The relationship design table should show **only** the tables and fields involved in the joins required for the query, hence, for this database, it will usually indicate some subset of the following relationships. This table may be blank if there is only one table needed for the query.

Some information about the given database:
Kimmy's Cakes is a relatively new specialty cake shop that provides custom cakes for special occasions or just for everyday dessert! Clients choose their cake type (e.g., vanilla, red velvet, lemon cream, etc.), size (e.g., small, medium, etc), the date by which they would like the cake to be ready, whether or not they want delivery (and where), and include any special instructions for the cake (such as writing to be included on the cake).

The price of a cake is determined by it's size (number of servings), the cost-to-make per serving for a particular type of cake, and the markup amount charged by the *Kimmy's Cakes*, which is 35%.

Each client can order more than one cake, but each order is for one cake only.

Below is the relationship diagram for this database:



With this understanding complete the design of the queries in **Part I** below.

Part I - Query Design

Design each of the following queries.

1. List all of the *Kimmy's Cake* orders that were placed before 2019. Your list should include the order number, the cake size, the date to be ready by, and whether or not the order was completed. Your list should be sorted from the oldest to most recent by the date to be ready.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

2. List all of the orders for extra large cakes that require delivery. Your list should include the order number, client ID, special instructions for the cake, and the delivery directions.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

3. List all cake orders for cakes with vanilla or chocolate pudding as the filling. Your list should include the cake name, the cost to make per serving, the date to be ready by, the special instructions for the cake and the flavour of the filling. Your list should be sorted by the cake name and, within each cake name, by date to be ready by (both in ascending order).

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

4. List the number of orders for each type of cake. Your list should include the cake type ID, the name of the cake, and the number of orders for each type of cake.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

5. A listing of all cake orders that do not need to be ready until after today. (**Hint:** Today's date is returned by the **Date()** function.) Your list should include the name of the client who ordered the cake (their first name, followed by their last name), the date the cake was ordered, the date that the cake needs to be ready by, the name of the cake and the name of the size of the cake ordered. Your list should be sorted by the last name of the client (in ascending order) and then by the first name of the client (also in ascending order).

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

6. List of the total number of servings ordered by each client over all cake orders. Your list should include the client ID and the total servings ordered by that client overall. The list should be sorted by the total servings ordered, from most to least.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

7. List of the average number of servings ordered for a particular cake type. The cake type ID should be given as a parameter to your query. Your list should include the cake type ID and the average number servings ordered for that cake type.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

8. List of all the orders placed in 2019. Include in your list, the order number, the name of the cake and the cake size (the two-digit code), as well as the price of the cake ordered and the number of days available in which to make the cake (i.e., the difference between dates for when the cake needs to be ready and when it was ordered). The price of a cake is calculated by multiplying the number of servings by the cost to make the cake (per serving), and then including a markup of 35% of the cost to make the cake.

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	**From Field**	**To Table**	**To Field**

9. Due to a rise in ingredient prices, *Kimmy's Cakes* has noticed that the cost to make each cake has increased by at least 10%. Create an **update query** to make this change.

Complete the Update Query Design Table:

Field:									
Table:									
Update To:									
Criteria:									
Or:									

10. List the total profit generated from all of the orders. Your list should result in a single number (the total profit) and assume the previously mentioned markup of 35% of the cost to make the cake. (**Hint:** The total profit from all orders should be $242.20.)

Complete the Query Design Table:

Field:									
Table:									
Total:									
Sort:									
Show(Y/N):									
Criteria:									
Or:									

Complete the Relationship Design Table (Join Table):

From Table	From Field	To Table	To Field

Part II - Implementation of Queries

In the file **COMP1600_A3_S19.accdb**, implement and run each of the queries designed in **Part I**. Make sure to name your queries with the corresponding query number, i.e., Query1, Query2, ..., Query10.

Submission

Submit your **COMP1600_A3_S19_DESIGN.docx** file and your **COMP1600_A3_S19.accdb** file containing your solutions using D2L's **Dropbox** tool. Note that the D2L **Dropbox** has been set up to allow you to submit updates to your assignment multiple times until the due date/time.