**COMP 2003 Relational databases**

**ASSIGNMENT 4**

Professor: Lyle Wood

Student name: FUYIJUN

Student ID: 200552722

**Part One:**

Create a database for this assignment then Use the a4\_data.sql file to populate your assignment database.

A screenshot of a computer

Description automatically generated

Show the table to prove that data were inserted into the table successfully.

A screenshot of a computer

Description automatically generated

1. Using a join \*do not use a subquery\*, please build a query that will return the first name,

last name and email of any software developers in my\_contacts.

I use the join function to match the IDs from two tables and filtered all software developer.

A screenshot of a computer

Description automatically generated

2. Using a subquery \*do not use a join\*, please build a query that will return the first name,

last name, and profession, of the contact who uses the email address

[tinfoilhat@conspiracy.ca](mailto:tinfoilhat@conspiracy.ca).

This query uses a subquery in the SELECT statement to retrieve the profession from the professions table based on the ID in the my\_contacts table. The WHERE clause filters the results to only include the contact with the specified email address. The SELECT statement then retrieves the first name, last name, and the result of the subquery as the profession.A screenshot of a computer

Description automatically generated

3. At your discretion, please use any combination of subqueries and joins to return the first

name, last name, profession and postal code of anyone living in Churchill, ON.

The main query selects the first name, last name, and the postal code from the my\_contacts table and uses a subquery to get the corresponding profession from the professions table.

It joins the my\_contacts table with the postal\_codes table using the postal\_code field.

The WHERE clause filters the results to only include rows where the city is 'Churchill, ON'.A screenshot of a computer

Description automatically generated

**Part Two:**

Using the table structures and relationships from *professions* and *my\_contacts* in the schema provided in Part One as an example, in Part Two you will create a table for *status* values and link those values to people in *my\_contacts* using a surrogate key.

1. Create a table called *status* selects and groups all *status* values from *my\_contacts* and stores

the unique values in the new table.

2. (If you need to) alter the new *status* table to contain a primary key – if you did this in the

last step you can skip to step 3.

Step1: Create the table and set status\_id as the primary key and the id will increment automatically.

A screenshot of a computer

Description automatically generated

Step 2: Find all the distinct status data from my contacts table.

A screenshot of a computer

Description automatically generated

Show the status table.

A screenshot of a computer

Description automatically generated

3. Use and UPDATE statement to make the *my\_contacts.status* column to contain the *ID* values

from the *status* table (rather than the status values as provided)

Before Q3, add status id into my-contacts table and set status id as a foreign key.

A screenshot of a computer

Description automatically generated

use JOIN to match the status from both tables.

A screenshot of a computer

Description automatically generated

Drop the old 'status' column from my\_contacts table

A screenshot of a computer

Description automatically generated

Change the column name to status to avoid duplicated column names.

A screenshot of a computer

Description automatically generated

4. After step 3 is successful, choose whether to *MODIFY* or *CHANGE* the *my\_contacts.status*

column to be an integer field because it is now storing status IDs rather than status values,

and do so. In your report, justify your choice of MODIFY or CHANGE.

Here should use MODIFY because I just change the name of the column. The data name of the column has already been changed before.

A screenshot of a computer

Description automatically generated

Show the table below:

A screenshot of a computer

Description automatically generated