**CSCE 4523 Database: Homework 3**

**Due: Friday, March 6, 2020, 11:59pm**

**All solutions must be prepared individually**

**Objective**

You are to implement a simple database using MySQL on turing.

**Program Description**

Create a database with the following schema:

**CLIENTS**

|  |  |  |
| --- | --- | --- |
| C\_ID | int | Primary Key |
| C\_NAME | char(50) | NOT NULL |
| C\_CITY | char(50) | NOT NULL |
| C\_ZIP | int | NOT NULL |

AGENTS

|  |  |  |
| --- | --- | --- |
| A\_ID | int | Primary Key |
| A\_NAME | char(50) | NOT NULL |
| A\_CITY | char(50) | NOT NULL |
| A\_ZIP | int | NOT NULL |

POLICY

|  |  |  |
| --- | --- | --- |
| POLICY\_ID | int | Primary Key |
| NAME | char(50) | NOT NULL |
| TYPE | char(50) | NOT NULL |
| COMMISION\_  PERCENTAGE | int | NOT NULL |

POLICIES\_SOLD

|  |  |  |
| --- | --- | --- |
| PURCHASE\_ID | int | Primary Key |
| AGENT\_ID | int | Foreign key references AGENTS(A\_ID) |
| CLIENT\_ID | int | Foreign key references CLIENTS(C\_ID) |
| POLICY\_ID | Int | Foreign key references POLICY(POLICY\_ID) |
| DATE\_PURCHASED | date | NOT NULL |
| AMOUNT | dec(6,2) | NOT NULL |

**Implementation**

Be sure to handle all entity integrity, referential integrity, and other constraints in your table definitions. Note: mysql does not enforce domain constraints, but include them in your definitions anyway so that, if you were to move to a different database or if they were to update their implementation, your database definition would be ready to go.   
Define your constraints such that: If rows in Agents, Clients, policy are attempted to be deleted, that operation should be blocked/disallowed.

Once the tables are created, insert records to match the data file at: <http://www.csce.uark.edu/~sgauch/4523/S20/hw/data.txt>

Hint: Type your queries into Notepad or Word or vim and cut and paste them to debug and then save them. Or, put them in a shell script (see posted example).

**What to Turn In**

To turn in your output, at the mysql prompt, type:

tee a3.log

Then execute the commands for tasks 1 through 10 to display the results on screen. When you are done, type:

notee

Upload a3.log to Blackboard.

### Create a Word or text report

#### Turn tee on and run the following within mysql

1. Show the tables that you have created. Use the command below:

SHOW TABLES;

2. Show the schema of each table. Use the command below, once per tablename:

DESC tablename;

3. Show how each table was created. Use the command below, once per tablename:

SHOW CREATE TABLE tablename;

4. Show the foreign keys of each table. Use the command (fill in your own db name):

SELECT COLUMN\_NAME, CONSTRAINT\_NAME, REFERENCED\_COLUMN\_NAME, REFERENCED\_TABLE\_NAME FROM INFORMATION\_SCHEMA.KEY\_COLUMN\_USAGE   
WHERE REFERENCED\_COLUMN\_NAME IS NOT NULL AND CONSTRAINT\_SCHEMA = 'your\_db';

5. Show all the records within every table. For each table, use the command:

SELECT \* FROM tablename;

#### For each of the following, output the query and the results of running the query.

6.A client named ’Taylor’ wants to see if there are any agents available in her city. Write a query to display agent id, agent name, agent city and agent zip of all available agents in Taylor’s city.

Note: Do not just plugin Taylors city manually by looking at the table, find Taylors city/zip using her name and then find agents in her city/zip.

7. Find out the second-highest commission percentage. Output the name of the policy, its type and the commission\_percentage associated with it.

8. Find out which type of policies have been sold in Fayetteville. Output the name of the policy, its type and number of times the policy has been sold.

9. Find out the most frequently sold policy in the year 2020 (based on no. of times the policy has been sold). Output the name of the policy, its type and its frequency as ‘COUNT’.

10. Using the Commission percentage we have for each policy in the policy table, find out how much each agent earned through commissions from selling the policies. Sort them in the decreasing order. Output the agent’s name and amount they earned from commissions.

#### Turn off tee (notee) and append the output to your report with the CREATE TABLE code. Turn this one document in.

**Grading:** Creating the tables with the correct constraints is worth 50 points. Running the SQL queries to insert and retrieve is also worth 50 points.

**Where to work:** This project must be done on turing.csce.uark.edu.