UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

School of Computer Science and Applied Mathematics
UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

COMS2002A: DATABASE FUNDAMENTALS

SEMESTER 1, 2024

LAB 2 – 20th February, 2024

Lecturer: Mr. Mikhail Chirkoot

2339279@students.wits.ac.za

DATA DEFINITION COMMANDS – ALTER, RENAME, TRUNCATE PART 1 - PRACTICE

This week, we'll continue with learning the Data Definition Commands. We'll learn how to use the ALTER, RENAME and TRUNCATE commands.

Note that you can refer to the file **Introduction to SQL - 2.pdf** file on Ulwazi for more details on the SQL queries.

1. First, let's create a table using the CREATE command we learnt in Lab 1. Create a table named *table_lab2*, with two columns – *first_name* and *age*.

   Type in the command:

   ```
   CREATE TABLE

   table_lab2( first_name

   VARCHAR(30), age INTEGER);
   ```

2. Now describe the table to get a view of the table structure. Type in the command:

   ```
   DESC table_lab2;
   ```

3. Next, we'll use the ALTER command to modify our table.

ALTER: All changes in table structure are made by using ALTER command.
ALTER command options include:

- ADD adds a column
- MODIFY changes column data types, data constraints
- DROP deletes a column

Let's see how to use the ALTER + ADD command to add a new column.

To add a new column to our existing table, type the following:

```
ALTER TABLE table_lab2

ADD COLUMN last_name VARCHAR(10);
```

4. Next, describe the table again to see the changes in the table structure. Run the following command to see the new structure of your table.

```
DESC table_lab2;
```

5. ALTER + MODIFY: Now let's modify the table. We can make changes to a column that already exists. Type in this command to change the last_name from VARCHAR(10) to VARCHAR(30).

```
ALTER TABLE table_lab2

MODIFY COLUMN last_name VARCHAR(30);
```

6. ALTER + CHANGE. Next, let's change a column name. Change the column name from *age* to *last_birthday_age* . Type in the following command.

```
ALTER TABLE table_lab2

CHANGE COLUMN age last_birthday_age INTEGER;
```

7. ALTER + DROP. We can also delete an existing column. Delete the last_birthday_age column by the command:
   `ALTER TABLE table_lab2 DROP last_birthday_age;`
8. We can also rename an existing table. Run these commands one after the other.

```
SHOW TABLES;
RENAME TABLE table_lab2 TO new_table_lab2;
SHOW TABLES;
```

   Now compare the output from the two SHOW TABLES commands to see the difference.

```
mysql> SHOW TABLES;
+--------------------+
| Tables_in_dolaperi |
+--------------------+
| my_first_table     |
| table_lab2         |
+--------------------+
2 rows in set (0.00 sec)

mysql> RENAME TABLE table_lab2 TO new_table_lab2;
Query OK, 0 rows affected (0.04 sec)

mysql> SHOW TABLES;
+--------------------+
| Tables_in_dolaperi |
+--------------------+
| my_first_table     |
| new_table_lab2     |
+--------------------+
2 rows in set (0.00 sec)
```

9. The TRUNCATE command can be used to remove all data from a table. It retains the structure of a table. To visualize how this command works, we first insert data into the table and then view (SELECT) the data. After executing the TRUNCATE command, all the data entered is deleted.
   Run the following commands:

   This command inserts a row of data into the table.
   ```
   INSERT INTO new_table_lab2 VALUES ('Olaperi', 'Okuboyejo');
   ```
   This command helps us to see the data in a table:
   ```
   SELECT * FROM new_table_lab2;
   ```

   This command deletes all the data in the table.
   ```
   TRUNCATE new_table_lab2;
   ```

   Run the SELECT command again and you'll see that the data entered is now deleted. It will return an Empty set.

   ```
   SELECT * FROM new_table_lab2;
   ```

```
mysql> INSERT INTO new_table_lab2 VALUES ('Olaperi', 'Okuboyejo');
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM new_table_lab2;
+------------+-----------+
| first_name | last_name |
+------------+-----------+
| Olaperi    | Okuboyejo |
+------------+-----------+
1 row in set (0.00 sec)

mysql> TRUNCATE new_table_lab2;
Query OK, 0 rows affected (0.06 sec)

mysql> SELECT * FROM new_table_lab2;
Empty set (0.00 sec)
```

PART 2 – YOUR TASK

Write the SQL commands to perform the following tasks. Upload all the files into the Gradescope assignment – **Lab 2.**

1. Create a table named **customer**. The column names and data types are is summarized in the following table. Save your SQL command for this task in a file called ***lab2_q1.txt***

ATTRIBUTE (FIELD NAME)	DATA TYPE
customer_id	CHAR(3)
customer_lname	VARCHAR(25)
customer_fname	VARCHAR(25)

2. Alter the table to add a new column. Call this new column ***customer_address*** with datatype **VARCHAR(255).** Save your SQL command for this task in a file called ***lab2_q2.txt***
3. Alter the table. Change the data type of the ***customer_id*** column into an INTEGER. Save your SQL command for this task in a file called ***lab2_q3.txt***
4. Alter the table. Change column ***customer_fname*** to ***first_name***, with the data_type as VARCHAR(25). Save your SQL command for this task in a file called ***lab2_q4.txt***
5. Alter the table. Drop column ***customer_lname***. Save your SQL command for this task in a file called ***lab2_q5.txt***
6. Rename the table name to ***new_customer***. Save your SQL command for this task in a file called ***lab2_q6.txt***