DBMS LAB ASSIGNMENT 09

Name: Taha Abbas
Class: BSCS-4A
Roll no: P200119
Date: APRIL 21 2022

Create a database name person and create the tables above.

```
MariaDB [(none)]> CREATE DATABASE PERSON;
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [(none)]> USE PERSON;
Database changed
MariaDB [PERSON]> CREATE TABLE USERS(
    -> USER_ID INT NOT NULL AUTO_INCREMENT,
    -> USER_NAME VARCHAR(20) NOT NULL,
    -> PASSWORD VARCHAR(20) NOT NULL,
    -> EMAIL varchar(20) NOT NULL,
    -> SALARY INT,
    -> PRIMARY KEY (USER_ID));
Query OK, 0 rows affected (0.025 sec)

MariaDB [PERSON]> CREATE TABLE SUMMARY(
    -> ID INT NOT NULL,
    -> TOTAL_USERS INT NOT NULL,
    -> YAHOO INT NOT NULL,
    -> HOTMAIL INT NOT NULL,
    -> GMAIL INT NOT NULL,
    -> PRIMARY KEY (ID));
Query OK, 0 rows affected (0.021 sec)
```

1. Write a procedure that take id, total_user, Yahoo, Hotmail, Gmail values as an input and insert the data into the table summary.

```
MariaDB [PERSON]> DELIMITER $$
MariaDB [PERSON]> CREATE PROCEDURE INSERT_INTO_SUMMARY(IN ID INT , IN TOTAL_USERS INT, IN YAHOO I
NT, IN HOTMAIL INT, IN GMAIL INT)
    -> BEGIN
    -> INSERT INTO SUMMARY(ID,TOTAL_USERS,YAHOO,HOTMAIL,GMAIL) VALUES (ID,TOTAL_USERS,YAHOO,HOTMA
IL,GMAIL);
    -> END $$
Query OK, 0 rows affected (0.012 sec)

MariaDB [PERSON]> CALL INSERT_INTO_SUMMARY(1,2,1,0,1) $$
Query OK, 1 row affected (0.010 sec)

MariaDB [PERSON]> SELECT * FROM SUMMARY $$
+----+-------------+-------+---------+-------+
| ID | TOTAL_USERS | YAHOO | HOTMAIL | GMAIL |
+----+-------------+-------+---------+-------+
| 1  |           2 |     1 |       0 |     1 |
+----+-------------+-------+---------+-------+
1 row in set (0.001 sec)
```

2. Write a procedure that take user_id, username, password, email values as an input and insert the data into the table Users.

```
MariaDB [PERSON]> CREATE PROCEDURE INSERT_INTO_USERS(IN USER_NAME VARCHAR(20), IN PASSWORD VARCHA
R(20), IN EMAIL VARCHAR(20), IN SALARY INT)
    -> BEGIN
    -> INSERT INTO USERS(USER_NAME,PASSWORD,EMAIL,SALARY) VALUES (USER_NAME,PASSWORD,EMAIL,SALARY
);
    -> END $$
MariaDB [PERSON]> CALL INSERT_INTO_USERS("ABC12","DEF321","ABC123@YAHOOO.COM",3000) $$
Query OK, 1 row affected (0.010 sec)

MariaDB [PERSON]> SELECT * FROM USERS $$
+---------+-----------+----------+--------------------+--------+
| USER_ID | USER_NAME | PASSWORD | EMAIL              | SALARY |
+---------+-----------+----------+--------------------+--------+
|       1 | ABC12     | DEF321   | ABC123@YAHOOO.COM  |   3000 |
+---------+-----------+----------+--------------------+--------+
1 row in set (0.001 sec)
```

```
MariaDB [PERSON]> CALL INSERT_INTO_USERS("XYZ789","ASD123","XYZ789@GMAIL.COM",5000) $$
Query OK, 1 row affected (0.011 sec)

MariaDB [PERSON]> SELECT * FROM USERS $$
+---------+-----------+----------+---------------------+--------+
| USER_ID | USER_NAME | PASSWORD | EMAIL               | SALARY |
+---------+-----------+----------+---------------------+--------+
|       1 | ABC12     | DEF321   | ABC123@YAHOOO.COM   |   3000 |
|       2 | XYZ789    | ASD123   | XYZ789@GMAIL.COM    |   5000 |
+---------+-----------+----------+---------------------+--------+
2 rows in set (0.001 sec)
```

3. Write a procedure that outputs the total number of records/rows in the users table.

```
MariaDB [PERSON]> CREATE PROCEDURE RECORDS(OUT TOTAL_RECORDS INT)
    -> BEGIN
    -> SELECT COUNT(USER_ID) INTO TOTAL_RECORDS FROM USERS;
    -> END $$
Query OK, 0 rows affected (0.011 sec)

MariaDB [PERSON]> CALL RECORDS(@TOTAL_RECORDS) $$
Query OK, 1 row affected (0.001 sec)

MariaDB [PERSON]> SELECT @TOTAL_RECORDS $$
+----------------+
| @TOTAL_RECORDS |
+----------------+
|              2 |
+----------------+
1 row in set (0.000 sec)
```

4. Write a procedure that outputs the average salary in the users table.

```
MariaDB [PERSON]> CREATE PROCEDURE AVG_SAL(OUT AVG_SALARY INT)
    -> BEGIN
    -> SELECT AVG(SALARY) INTO AVG_SALARY FROM USERS;
    -> END $$
Query OK, 0 rows affected (0.011 sec)

MariaDB [PERSON]> CALL AVG_SAL(@AVG_SALARY) $$
Query OK, 1 row affected (0.001 sec)

MariaDB [PERSON]> SELECT @AVG_SALARY $$
+-------------+
| @AVG_SALARY |
+-------------+
|        4000 |
+-------------+
1 row in set (0.000 sec)
```

5. Create a procedure named minmax which displays the lowest and the highest salary of the users. Run the procedure to display the lowest and the highest salary.

```
MariaDB [PERSON]> CREATE PROCEDURE MINMAX_SAL(OUT MIN_SAL INT, OUT MAX_SAL INT)
    -> BEGIN
    -> SELECT MIN(SALARY),MAX(SALARY) INTO MIN_SAL, MAX_SAL FROM USERS;
    -> END $$
Query OK, 0 rows affected (0.011 sec)

MariaDB [PERSON]> CALL MINMAX_SAL(@MIN_SAL, @MAX_SAL) $$
Query OK, 1 row affected (0.001 sec)

MariaDB [PERSON]> SELECT @MIN_SAL, @MAX_SAL $$
+----------+----------+
| @MIN_SAL | @MAX_SAL |
+----------+----------+
|     3000 |     5000 |
+----------+----------+
1 row in set (0.000 sec)
```

6. Create trigger(s) to update the records in the summary table. The summary table will contain only one record and will be updated each time a new entry is made to the users table, or a user is deleted from the users table. You need to create the following two triggers.

I. Whenever a new entry is made to the user's table then total_user attribute in the summary table must be incremented by 1.

2. Whenever a record is deleted from the user's table then total_user attribute in the summary table must be decremented by 1.

ADDING TUPLES IN THE TABLE

```
MariaDB [PERSON]> CALL INSERT_INTO_USERS("233JKL","WOO123","233JKL@GMAIL.COM",7000)$$
Query OK, 1 row affected (0.009 sec)

MariaDB [PERSON]> CALL INSERT_INTO_USERS("TUS110","PQR456","TUS110@GMAIL.COM",9000)$$
Query OK, 1 row affected (0.002 sec)

MariaDB [PERSON]> CALL INSERT_INTO_USERS("CSV567","CNN565","CSV567@HOTMAIL.COM",1000)$$
Query OK, 1 row affected (0.009 sec)
```

CREATING TRIGGER FOR INSERTION

```
MariaDB [PERSON]> CREATE TRIGGER USER_INSERTION AFTER INSERT ON PERSON.USERS
    -> FOR EACH ROW BEGIN
    -> UPDATE SUMMARY SET TOTAL_USERS = TOTAL_USERS + 1 WHERE ID = 1;
    -> END $$
Query OK, 0 rows affected (0.013 sec)

MariaDB [PERSON]> CALL INSERT_INTO_USERS("233JKL","WOO123","233JKL@GMAIL.COM",7000) $$
Query OK, 2 rows affected (0.009 sec)
```

```
MariaDB [PERSON]> SELECT * FROM USERS $$
+---------+-----------+----------+----------------------+--------+
| USER_ID | USER_NAME | PASSWORD | EMAIL                | SALARY |
+---------+-----------+----------+----------------------+--------+
|       1 | ABC123    | DEF321   | ABC123@YAHOOO.COM    |   3000 |
|       2 | XYZ789    | ASD123   | XYZ789@GMAIL.COM     |   5000 |
|       3 | JKL233    | WOO123   | JKL233@GMAIL.COM     |   7000 |
|       4 | TUS110    | PQR456   | TUS110@GMAIL.COM     |   9000 |
|       5 | CSV567    | CNN565   | CSV567@HOTMAIL.COM   |   1000 |
|       6 | MSM346    | WER990   | MSM436@YAHOO.COM     |   7500 |
+---------+-----------+----------+----------------------+--------+
6 rows in set (0.001 sec)
```

CREATING TRIGGER FOR DELETION

```
MariaDB [PERSON]> CREATE TRIGGER USER_DELETION AFTER DELETE ON PERSON.USERS
    -> FOR EACH ROW BEGIN
    -> UPDATE SUMMARY SET TOTAL_USERS = TOTAL_USERS -1 WHERE ID = 1;
    -> END $$
Query OK, 0 rows affected (0.006 sec)

MariaDB [PERSON]> DELETE FROM USERS WHERE USER_ID = 3;
    -> $$
Query OK, 1 row affected (0.010 sec)

MariaDB [PERSON]> SELECT * FROM SUMMARY;
    -> $$
+----+-------------+-------+---------+-------+
| ID | TOTAL_USERS | YAHOO | HOTMAIL | GMAIL |
+----+-------------+-------+---------+-------+
|  1 |           4 |     1 |       0 |     1 |
+----+-------------+-------+---------+-------+
1 row in set (0.001 sec)
```