A6

Name: Pranay Shankarrao Chavhan

Roll No: 31412

Problem Statement:- Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table Cust_New with the data available in the table Cust_Old. If the data in the first table already exist in the second table then that data should be skipped.

mysql> CREATE TABLE Cust_New (

->   id INT AUTO_INCREMENT PRIMARY KEY,

->   name VARCHAR(255)

->);

Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO Cust_New (name) VALUES

->   ('Rahul'),

->   ('Priya'),

->   ('Amit'),

->   ('Anjali'),

->   ('Vikram');

Query OK, 5 rows affected (0.02 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> CREATE TABLE Cust_Old (

->   id INT AUTO_INCREMENT PRIMARY KEY,

->   name VARCHAR(255)

->);

Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO Cust_Old (name) VALUES

->   ('Anjali'),

```
->     ('Kiran'),
->     ('Rahul');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> DELIMITER //
mysql> CREATE PROCEDURE MergeData()
-> BEGIN
->     DECLARE done INT DEFAULT 0;
->     DECLARE cust_new_value VARCHAR(255);
->

->     DECLARE cust_cursor CURSOR FOR
->         SELECT name FROM Cust_New;
->

->     DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
->
->     OPEN cust_cursor;
->
->     read_loop: LOOP
->         FETCH cust_cursor INTO cust_new_value;
->
->         IF done THEN
->             LEAVE read_loop;
->         END IF;
->
->         IF NOT EXISTS (SELECT * FROM Cust_Old WHERE name = cust_new_value)
THEN
->             INSERT INTO Cust_Old (name) VALUES (cust_new_value);
->         END IF;
->     END LOOP;
->
->     CLOSE cust_cursor;
-> END //
Query OK, 0 rows affected (0.01 sec)
```

mysql> DELIMITER ;

mysql> CALL MergeData();

Query OK, 0 rows affected (0.01 sec)

```
mysql> select * from Cust_Old;
+----+--------+
| id | name   |
+----+--------+
|  1 | Anjali |
|  2 | Kiran  |
|  3 | Rahul  |
|  4 | Priya  |
|  5 | Amit   |
|  6 | Vikram |
+----+--------+
6 rows in set (0.00 sec)
```