

Team ID: 216_226_227_244

QUESTION 3:

```
216_226_227_244>create database fresh_bite;
Query OK, 1 row affected (0.01 sec)

216_226_227_244>-- Create the database
216_226_227_244>USE fresh_bite;
Database changed
216_226_227_244>
216_226_227_244>-- Customer Table
216_226_227_244>CREATE TABLE customer (
    ->     customer_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     name VARCHAR(100) NOT NULL,
    ->     email VARCHAR(100) NOT NULL UNIQUE,
    ->     phone_number VARCHAR(15) NOT NULL,
    ->     address VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.02 sec)

216_226_227_244>
216_226_227_244>-- Restaurant Table
216_226_227_244>CREATE TABLE restaurant (
    ->     restaurant_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     name VARCHAR(100) NOT NULL,
    ->     location VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.02 sec)

216_226_227_244>
216_226_227_244>-- Menu Table
216_226_227_244>CREATE TABLE menu (
    ->     menu_item_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     restaurant_id INT NOT NULL,
    ->     item_name VARCHAR(100) NOT NULL,
    ->     price DECIMAL(10, 2) NOT NULL,
    ->     availability BOOLEAN NOT NULL,
    ->     FOREIGN KEY (restaurant_id) REFERENCES restaurant(restaurant_id)

216_226_227_244>
216_226_227_244>-- Order Table
216_226_227_244>CREATE TABLE `order` (
    ->     order_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     customer_id INT NOT NULL,
    ->     restaurant_id INT NOT NULL,
    ->     total_price DECIMAL(10, 2) NOT NULL,
    ->     order_status VARCHAR(50) NOT NULL,
    ->     order_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    ->     FOREIGN KEY (customer_id) REFERENCES customer(customer_id),
    ->     FOREIGN KEY (restaurant_id) REFERENCES restaurant(restaurant_id)
    -> );
Query OK, 0 rows affected (0.04 sec)

216_226_227_244>
216_226_227_244>-- Order Items Table
216_226_227_244>CREATE TABLE order_item (
    ->     order_item_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     order_id INT NOT NULL,
    ->     menu_item_id INT NOT NULL,
    ->     quantity INT NOT NULL,
    ->     price DECIMAL(10, 2) NOT NULL,
    ->     FOREIGN KEY (order_id) REFERENCES `order`(order_id),
    ->     FOREIGN KEY (menu_item_id) REFERENCES menu(menu_item_id)
    -> );
Query OK, 0 rows affected (0.03 sec)

216_226_227_244>
216_226_227_244>-- Delivery Table
216_226_227_244>CREATE TABLE delivery (
    ->     delivery_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     order_id INT NOT NULL,
    ->     delivery_person VARCHAR(100) NOT NULL,
    ->     delivery_status VARCHAR(50) NOT NULL,
    ->     delivery_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    ->     FOREIGN KEY (order_id) REFERENCES `order`(order_id)
    -> );
```

```

216_226_227_244>
216_226_227_244>-- Review Table
216_226_227_244>CREATE TABLE review (
->     review_id INT PRIMARY KEY AUTO_INCREMENT,
->     customer_id INT NOT NULL,
->     restaurant_id INT NOT NULL,
->     rating INT NOT NULL CHECK (rating BETWEEN 1 AND 5),
->     review_text TEXT,
->     review_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
->     FOREIGN KEY (customer_id) REFERENCES customer(customer_id),
->     FOREIGN KEY (restaurant_id) REFERENCES restaurant(restaurant_id)
-> );
Query OK, 0 rows affected (0.02 sec)

216_226_227_244>
216_226_227_244>
216_226_227_244>SELECT * FROM customer;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM restaurant;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM menu;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM `order`;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM order_item;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM delivery;
Empty set (0.00 sec)

216_226_227_244>SELECT * FROM review;
Empty set (0.00 sec)

```

```

216_226_227_244>
216_226_227_244>
216_226_227_244>-- Sample data insertion for each table
216_226_227_244>-- Customers
216_226_227_244>INSERT INTO customer (name, email, phone_number, address) VALUES
-> ('John Doe', 'john@email.com', '1234567890', '123 Main St'),
-> ('Jane Smith', 'jane@email.com', '12345678901', '456 Oak Ave'),
-> ('Bob Wilson', 'bob@email.com', '3456789012', '789 Pine Rd'),
-> ('Alice Brown', 'alice@email.com', '4567890123', '321 Maple Dr'),
-> ('Charlie Davis', 'charlie@email.com', '5678901234', '654 Cedar Ln');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

```

```

216_226_227_244>
216_226_227_244>-- Restaurants
216_226_227_244>INSERT INTO restaurant (name, location) VALUES
-> ('Spice Palace', '100 Food Court'),
-> ('Burger Heaven', '200 Fast Lane'),
-> ('Pizza Paradise', '300 Italian Way'),
-> ('Sushi Supreme', '400 Ocean Drive'),
-> ('Veggie Delight', '500 Green Street');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

```

```

216_226_227_244>
216_226_227_244>-- Menu Items
216_226_227_244>INSERT INTO menu (restaurant_id, item_name, price, availability) VALUES
-> (1, 'Butter Chicken', 15.99, true),
-> (1, 'Naan Bread', 3.99, true),
-> (2, 'Classic Burger', 12.99, true),
-> (2, 'French Fries', 4.99, true),
-> (3, 'Margherita Pizza', 18.99, true);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

```

```

216_226_227_244>
216_226_227_244>-- Orders
216_226_227_244>INSERT INTO `order` (customer_id, restaurant_id, total_price, order_status, order_time) VALUES
-> (1, 1, 35.98, 'Delivered', '2024-03-15 12:00:00'),
-> (2, 2, 22.97, 'Placed', '2024-03-15 12:30:00'),
-> (3, 3, 18.99, 'Cancelled', '2024-03-15 13:00:00'),
-> (4, 1, 19.98, 'Delivered', '2024-03-15 13:30:00'),
-> (5, 2, 17.98, 'Placed', '2024-03-15 14:00:00');
Query OK, 5 rows affected (0.00 sec)
Records: 5 Duplicates: 0 Warnings: 0

216_226_227_244>
216_226_227_244>-- Order Items
216_226_227_244>INSERT INTO order_item (order_id, menu_item_id, quantity, price) VALUES
-> (1, 1, 2, 31.98),
-> (1, 2, 1, 3.99),
-> (2, 3, 1, 12.99),
-> (2, 4, 2, 9.98),
-> (3, 5, 1, 18.99);
Query OK, 5 rows affected (0.00 sec)
Records: 5 Duplicates: 0 Warnings: 0

216_226_227_244>
216_226_227_244>-- Deliveries
216_226_227_244>INSERT INTO delivery (order_id, delivery_person, delivery_status, delivery_time) VALUES
-> (1, 'Mike Johnson', 'Delivered', '2024-03-15 12:45:00'),
-> (2, 'Sarah Williams', 'Assigned', '2024-03-15 13:00:00'),
-> (3, 'Tom Brown', 'In Progress', '2024-03-15 13:30:00'),
-> (4, 'Lisa Davis', 'Delivered', '2024-03-15 14:15:00'),
-> (5, 'David Wilson', 'Assigned', '2024-03-15 14:30:00');
Query OK, 5 rows affected (0.00 sec)
Records: 5 Duplicates: 0 Warnings: 0

```

```

216_226_227_244>
216_226_227_244>-- Reviews
216_226_227_244>INSERT INTO review (customer_id, restaurant_id, rating, review_text) VALUES
-> (1, 1, 5, 'Excellent food and service!'),
-> (2, 2, 4, 'Good burger, will order again'),
-> (3, 3, 5, 'Best pizza in town'),
-> (4, 1, 4, 'Tasty food but delivery was a bit late'),
-> (5, 2, 3, 'Decent food, could be better');
Query OK, 5 rows affected (0.00 sec)
Records: 5 Duplicates: 0 Warnings: 0

```

```

216_226_227_244>
216_226_227_244>
216_226_227_244>
216_226_227_244>SELECT * FROM customer;
+-----+-----+-----+-----+-----+
| customer_id | name      | email        | phone_number | address      |
+-----+-----+-----+-----+-----+
| 1 | John Doe | john@email.com | 1234567890 | 123 Main St |
| 2 | Jane Smith | jane@email.com | 2345678901 | 456 Oak Ave |
| 3 | Bob Wilson | bob@email.com | 3456789012 | 789 Pine Rd |
| 4 | Alice Brown | alice@email.com | 4567890123 | 321 Maple Dr |
| 5 | Charlie Davis | charlie@email.com | 5678901234 | 654 Cedar Ln |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

216_226_227_244>SELECT * FROM restaurant;
+-----+-----+-----+
| restaurant_id | name      | location      |
+-----+-----+-----+
| 1 | Spice Palace | 100 Food Court |
| 2 | Burger Heaven | 200 Fast Lane |
| 3 | Pizza Paradise | 300 Italian Way |
| 4 | Sushi Supreme | 400 Ocean Drive |
| 5 | Veggie Delight | 500 Green Street |
+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

216_226_227_244>SELECT * FROM menu;
+-----+-----+-----+-----+-----+
| menu_item_id | restaurant_id | item_name | price | availability |
+-----+-----+-----+-----+-----+
| 1 | 1 | Butter Chicken | 15.99 | 1 |
| 2 | 1 | Naan Bread | 3.99 | 1 |
| 3 | 2 | Classic Burger | 12.99 | 1 |
| 4 | 2 | French Fries | 4.99 | 1 |
| 5 | 3 | Margherita Pizza | 18.99 | 1 |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

216_226_227_244>SELECT * FROM `order`;
+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 35.98 | Delivered | 2024-03-15 12:00:00 |
| 2 | 2 | 2 | 22.97 | Placed | 2024-03-15 12:30:00 |
| 3 | 3 | 3 | 18.99 | Cancelled | 2024-03-15 13:00:00 |
| 4 | 4 | 1 | 19.98 | Delivered | 2024-03-15 13:30:00 |
| 5 | 5 | 2 | 17.98 | Placed | 2024-03-15 14:00:00 |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

216_226_227_244>SELECT * FROM order_item;
+-----+-----+-----+-----+-----+
| order_item_id | order_id | menu_item_id | quantity | price |
+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 2 | 31.98 |
| 2 | 1 | 2 | 1 | 3.99 |
| 3 | 2 | 3 | 1 | 12.99 |
| 4 | 2 | 4 | 2 | 9.98 |
| 5 | 3 | 5 | 1 | 18.99 |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

216_226_227_244>SELECT * FROM delivery;
+-----+-----+-----+-----+
| delivery_id | order_id | delivery_person | delivery_status | delivery_time |
+-----+-----+-----+-----+
| 1 | 1 | Mike Johnson | Delivered | 2024-03-15 12:45:00 |
| 2 | 2 | Sarah Williams | Assigned | 2024-03-15 13:00:00 |
| 3 | 3 | Tom Brown | In Progress | 2024-03-15 13:30:00 |
| 4 | 4 | Lisa Davis | Delivered | 2024-03-15 14:15:00 |
| 5 | 5 | David Wilson | Assigned | 2024-03-15 14:30:00 |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)

216_226_227_244>SELECT * FROM review;
+-----+-----+-----+-----+-----+-----+
| review_id | customer_id | restaurant_id | rating | review_text | review_time |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 5 | Excellent food and service! | 2024-11-18 10:19:38 |
| 2 | 2 | 2 | 4 | Good burger, will order again | 2024-11-18 10:19:38 |
| 3 | 3 | 3 | 5 | Best pizza in town | 2024-11-18 10:19:38 |
| 4 | 4 | 1 | 4 | Tasty food but delivery was a bit late | 2024-11-18 10:19:38 |
| 5 | 5 | 2 | 3 | Decent food, could be better | 2024-11-18 10:19:38 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

QUESTION 4:

Server:

```
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.39 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> prompt 216_226_227_244>
PROMPT set to '216_226_227_244>'
216_226_227_244>create user 'PES1UG22CS226'@'%' identified by 'YASG@123';
Query OK, 0 rows affected (0.04 sec)

216_226_227_244>create database fresh_bite;
Query OK, 1 row affected (0.04 sec)

216_226_227_244>grant all privileges on fresh_bite.* to 'PES1UG22CS226'@'%';
Query OK, 0 rows affected (0.03 sec)
```

Client:

```
~
↳ sudo mysql -u PES1UG22CS226 -p -h 192.168.81.214 -P 3306 fresh_bite
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.39 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> prompt 216_226_227_244>
PROMPT set to '216_226_227_244>'
216_226_227_244>show tables;
+-----+
| Tables_in_fresh_bite |
+-----+
| customer             |
| delivery              |
| menu                 |
| order                 |
| order_item            |
| restaurant            |
| review                |
+-----+
7 rows in set (0.21 sec)
```

Client:

```
216_226_227_244>-- First delete related records in child tables
216_226_227_244>DELETE FROM delivery
    -> WHERE order_id IN (SELECT order_id FROM `order` WHERE order_status = 'Cancelled');
Query OK, 1 row affected (0.46 sec)

216_226_227_244>
216_226_227_244>DELETE FROM order_item
    -> WHERE order_id IN (SELECT order_id FROM `order` WHERE order_status = 'Cancelled');
Query OK, 1 row affected (0.08 sec)

216_226_227_244>
216_226_227_244>-- Then delete from the main order table
216_226_227_244>DELETE FROM `order`
    -> WHERE order_status = 'Cancelled';
Query OK, 1 row affected (0.24 sec)

216_226_227_244>select * from `orders`;
ERROR 1146 (42S02): Table 'fresh_bite.orders' doesn't exist
216_226_227_244>select * from `order`;
+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+
|      1 |          1 |            1 |      35.98 | Delivered   | 2024-03-15 12:00:00 |
|      2 |          2 |            2 |      22.97 | Placed     | 2024-03-15 12:30:00 |
|      4 |          4 |            1 |      19.98 | Delivered   | 2024-03-15 13:30:00 |
|      5 |          5 |            2 |      17.98 | Placed     | 2024-03-15 14:00:00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.82 sec)
```

Server:

```
216_226_227_244>select * from `order`;
+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+
|      1 |          1 |            1 |      35.98 | Delivered   | 2024-03-15 12:00:00 |
|      2 |          2 |            2 |      22.97 | Placed     | 2024-03-15 12:30:00 |
|      4 |          4 |            1 |      19.98 | Delivered   | 2024-03-15 13:30:00 |
|      5 |          5 |            2 |      17.98 | Placed     | 2024-03-15 14:00:00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

216_226_227_244>
```

QUESTION 5:

```
216_226_227_244>WITH RECURSIVE CustomerRestaurants AS (
->      -- Base case: Get all distinct customer-restaurant combinations
->      SELECT
->          customer_id,
->          COUNT(DISTINCT restaurant_id) as restaurant_count
->      FROM `order`
->      GROUP BY customer_id
->  )
->  SELECT
->      c.name,
->      COALESCE(cr.restaurant_count, 0) as unique_restaurants
->  FROM customer c
->  LEFT JOIN CustomerRestaurants cr ON c.customer_id = cr.customer_id
->  ORDER BY unique_restaurants DESC;
+-----+-----+
| name      | unique_restaurants |
+-----+-----+
| John Doe   |              1 |
| Jane Smith |              1 |
| Alice Brown |             1 |
| Charlie Davis |           1 |
| Bob Wilson   |              0 |
+-----+-----+
5 rows in set (0.07 sec)

216_226_227_244>
```

QUESTION 6:

```
216_226_227_244>DELIMITER //
216_226_227_244>
216_226_227_244>CREATE PROCEDURE transfer_delivery(
->      IN p_order_id INT,
->      IN p_old_agent VARCHAR(100),
->      IN p_new_agent VARCHAR(100)
-> )
-> BEGIN
->     DECLARE v_delivery_status VARCHAR(50);
->
->     -- Check if order exists and is not delivered
->     SELECT delivery_status INTO v_delivery_status
->     FROM delivery
->     WHERE order_id = p_order_id AND delivery_person = p_old_agent;
->
->     IF v_delivery_status IS NULL THEN
->         SIGNAL SQLSTATE '45000'
->         SET MESSAGE_TEXT = 'Order not found or agent mismatch';
->     ELSEIF v_delivery_status = 'Delivered' THEN
->         SIGNAL SQLSTATE '45000'
->         SET MESSAGE_TEXT = 'Cannot transfer delivered orders';
->     ELSE
->         -- Update the delivery record with new agent and status
->         UPDATE delivery
->             SET delivery_person = p_new_agent,
->                 delivery_status = CONCAT('Reassigned from ', p_old_agent),
->                 delivery_time = CURRENT_TIMESTAMP
->             WHERE order_id = p_order_id
->                 AND delivery_person = p_old_agent;
->
->         SELECT CONCAT('Order ', p_order_id, ' successfully transferred from ',
->                     p_old_agent, ' to ', p_new_agent) AS message;
->     END IF;
-> END //
Query OK, 0 rows affected (0.42 sec)
```

```
216_226_227_244>SELECT * FROM delivery;
+-----+-----+-----+-----+
| delivery_id | order_id | delivery_person | delivery_status | delivery_time |
+-----+-----+-----+-----+
| 1 | 1 | Mike Johnson | Delivered | 2024-03-15 12:45:00 |
| 2 | 2 | Mike Johnson | Reassigned | 2024-03-15 13:00:00 |
| 4 | 4 | Lisa Davis | Delivered | 2024-03-15 14:15:00 |
| 5 | 5 | David Wilson | Assigned | 2024-03-15 14:30:00 |
+-----+-----+-----+-----+
4 rows in set (0.37 sec)
```

```
216_226_227_244>CALL transfer_delivery(5, 'David Wilson', 'Akash');
+-----+
| message |
+-----+
| Order 5 successfully transferred from David Wilson to Akash |
+-----+
1 row in set (0.33 sec)
```

```
Query OK, 0 rows affected (0.33 sec)
```

```
216_226_227_244>SELECT * FROM delivery WHERE order_id = 5;
+-----+-----+-----+-----+
| delivery_id | order_id | delivery_person | delivery_status | delivery_time |
+-----+-----+-----+-----+
| 5 | 5 | Akash | Reassigned from David Wilson | 2024-11-18 11:09:11 |
+-----+-----+-----+-----+
1 row in set (0.69 sec)
```

```
216_226_227_244>■
```

QUESTION 7:

```
216_226_227_244>DELIMITER //
216_226_227_244>CREATE TRIGGER prevent_incomplete_order_reviews
-> BEFORE INSERT ON review
-> FOR EACH ROW
-> BEGIN
->     DECLARE order_count INT;
->
->     -- Check if there exists a delivered order for this customer and restaurant
->     SELECT COUNT(*) INTO order_count
->     FROM `order`
->     WHERE customer_id = NEW.customer_id
->     AND restaurant_id = NEW.restaurant_id
->     AND order_status = 'Delivered';
->
->     -- If no delivered order exists, raise an error
->     IF order_count = 0 THEN
->         SIGNAL SQLSTATE '45000'
->         SET MESSAGE_TEXT = 'Reviews can only be submitted for completed orders';
->     END IF;
-> END///
Query OK, 0 rows affected (0.09 sec)

216_226_227_244>
216_226_227_244>DELIMITER ;
```

localhost:8501

FreshBite Review System

Select Customer

Select Restaurant

Rating

5

1

Review

this is a test

Reviews can only be submitted for completed orders

Existing Reviews

```

Database changed
216_226_227_244>select * from `order`;
+-----+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 35.98 | Delivered | 2024-03-15 12:00:00 |
| 2 | 2 | 2 | 22.97 | Placed | 2024-03-15 12:30:00 |
| 4 | 4 | 1 | 19.98 | Delivered | 2024-03-15 13:30:00 |
| 5 | 5 | 2 | 17.98 | Placed | 2024-03-15 14:00:00 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

216_226_227_244>select * from customer;
+-----+-----+-----+-----+-----+
| customer_id | name | email | phone_number | address |
+-----+-----+-----+-----+-----+
| 1 | John Doe | john@email.com | 1234567890 | 123 Main St |
| 2 | Jane Smith | jane@email.com | 2345678901 | 456 Oak Ave |
| 3 | Bob Wilson | bob@email.com | 3456789012 | 789 Pine Rd |
| 4 | Alice Brown | alice@email.com | 4567890123 | 321 Maple Dr |
| 5 | Charlie Davis | charlie@email.com | 5678901234 | 654 Cedar Ln |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

216_226_227_244>select * from restaurant;
+-----+-----+
| restaurant_id | name | location |
+-----+-----+
| 1 | Spice Palace | 100 Food Court |
| 2 | Burger Heaven | 200 Fast Lane |
| 3 | Pizza Paradise | 300 Italian Way |
| 4 | Sushi Supreme | 400 Ocean Drive |
| 5 | Veggie Delight | 500 Green Street |
+-----+-----+
5 rows in set (0.00 sec)

```

C ⓘ localhost:8501

FreshBite Review System

Select Customer

Select Restaurant

Rating

Review

this is great

Review submitted successfully!

Existing Reviews

```

216_226_227_224>select * from review;
+-----+-----+-----+-----+-----+
| review_id | customer_id | restaurant_id | rating | review_text |
+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 5 | Excellent food and service! |
| 2 | 2 | 2 | 4 | Good burger, will order again |
| 3 | 3 | 3 | 5 | Best pizza in town |
| 4 | 4 | 1 | 4 | Tasty food but delivery was a bit late |
| 5 | 5 | 2 | 3 | Decent food, could be better |
| 9 | 1 | 1 | 5 | this is great |
+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

```

QUESTION 8:

```

216_226_227_224> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

216_226_227_224>
216_226_227_224> -- T1 locks the row with order_id = 4
216_226_227_224> SELECT * FROM `order` WHERE order_id = 4 FOR UPDATE;
+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+
| 4 | 4 | 1 | -130.02 | Delivered | 2024-03-15 13:30:00 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

216_226_227_224>
216_226_227_224> -- Correct way to update the total_price column
216_226_227_224> UPDATE `order`
-> SET total_price = total_price + 100
-> WHERE order_id = 4;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

216_226_227_224>
216_226_227_224> -- Simulate delay in releasing the lock
216_226_227_224> DO SLEEP(30);
Query OK, 0 rows affected (30.01 sec)

216_226_227_224>
216_226_227_224> -- Commit the transaction (unlocking the rows)
216_226_227_224> COMMIT;
Query OK, 0 rows affected (0.00 sec)

216_226_227_224> select * from `order`;
+-----+-----+-----+-----+-----+
| order_id | customer_id | restaurant_id | total_price | order_status | order_time |
+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | 35.98 | Delivered | 2024-03-15 12:00:00 |
| 2 | 2 | 2 | 22.97 | Placed | 2024-03-15 12:30:00 |
| 4 | 4 | 1 | -80.02 | Delivered | 2024-03-15 13:30:00 |
| 5 | 5 | 2 | 17.98 | Placed | 2024-03-15 14:00:00 |
| 6 | 1 | 1 | 600.00 | Placed | 2024-11-18 12:29:16 |
| 7 | 1 | 1 | 700.00 | Placed | 2024-11-18 12:36:00 |
+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

216_226_227_224> |

```

```

216_226_227_244>-- Simulate Transaction T2
216_226_227_244>START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

216_226_227_244>
216_226_227_244>-- First UPDATE in T2 (Fails due to lock timeout)
216_226_227_244>UPDATE `order`
-> SET total_price = total_price - 50
-> WHERE order_id = 4;
Query OK, 1 row affected (31.91 sec)
Rows matched: 1  Changed: 1  Warnings: 0

216_226_227_244>commit;
Query OK, 0 rows affected (0.00 sec)

216_226_227_244>-- Simulate Transaction T2
216_226_227_244>START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

216_226_227_244>
216_226_227_244>-- First UPDATE in T2 (Fails due to lock timeout)
216_226_227_244>UPDATE `order`
-> SET total_price = total_price - 50
-> WHERE order_id = 4;
Query OK, 1 row affected (15.37 sec)
Rows matched: 1  Changed: 1  Warnings: 0

216_226_227_244>COMMIT;
Query OK, 0 rows affected (0.00 sec)

216_226_227_244>

```

QUESTION 9:

```

216_226_227_224> -- First, create the function
216_226_227_224> DELIMITER //
216_226_227_224>
216_226_227_224> CREATE FUNCTION restaurant_rating_summary(p_restaurant_id INT)
-> RETURNS VARCHAR(255)
-> READS SQL DATA
-> BEGIN
->     DECLARE total_ratings INT;
->     DECLARE avg_rating DECIMAL(4,2);
->     DECLARE five_star_percent DECIMAL(4,1);
->
->     -- Get total ratings
->     SELECT COUNT(*) INTO total_ratings
->     FROM review
->     WHERE restaurant_id = p_restaurant_id;
->
->     IF total_ratings = 0 THEN
->         RETURN NULL;
->     END IF;
->
->     -- Calculate average rating
->     SELECT ROUND(AVG(rating), 2) INTO avg_rating
->     FROM review
->     WHERE restaurant_id = p_restaurant_id;
->
->     -- Calculate percentage of 5-star ratings
->     SELECT ROUND(COUNT(*) * 100.0 / total_ratings, 1) INTO five_star_percent
->     FROM review
->     WHERE restaurant_id = p_restaurant_id
->     AND rating = 5;
->
->     RETURN CONCAT(
->         'Total Ratings: ', total_ratings,
->         ', Average Rating: ', avg_rating,
->         ', 5-Star Percentage: ', five_star_percent, '%'
->     );
-> END //
Query OK, 0 rows affected (0.04 sec)

216_226_227_224>
216_226_227_224> DELIMITER ;
216_226_227_224>
216_226_227_224> -- Now, let's call the function in different ways:
216_226_227_224>
216_226_227_224> -- 1. Test for a single restaurant with reviews
216_226_227_224> SELECT
->     r.name AS restaurant_name,
->     restaurant_rating_summary(r.restaurant_id) AS rating_summary
->     FROM restaurant r
->     WHERE r.restaurant_id = 1;

```

```

+-----+
| restaurant_name | rating_summary |
+-----+
| Spice Palace   | Total Ratings: 3, Average Rating: 4.67, 5-Star Percentage: 66.7% |
+-----+
1 row in set (0.03 sec)

216_226_227_224>
216_226_227_224> -- 2. Test for all restaurants to see their summaries
216_226_227_224> SELECT
->     r.restaurant_id,
->     r.name AS restaurant_name,
->     restaurant_rating_summary(r.restaurant_id) AS rating_summary
-> FROM restaurant r;
+-----+
| restaurant_id | restaurant_name | rating_summary |
+-----+
|      1 | Spice Palace   | Total Ratings: 3, Average Rating: 4.67, 5-Star Percentage: 66.7% |
|      2 | Burger Heaven  | Total Ratings: 2, Average Rating: 3.50, 5-Star Percentage: 0.0% |
|      3 | Pizza Paradise | Total Ratings: 1, Average Rating: 5.00, 5-Star Percentage: 100.0% |
|      4 | Sushi Supreme  | NULL           |
|      5 | Veggie Delight | NULL           |
+-----+
5 rows in set (0.00 sec)

216_226_227_224>
216_226_227_224> -- 3. Verification query to double-check the results
216_226_227_224> SELECT
->     r.restaurant_id,
->     r.name,
->     COUNT(rev.review_id) as total_ratings,
->     ROUND(AVG(rev.rating), 2) as avg_rating,
->     ROUND(SUM(CASE WHEN rev.rating = 5 THEN 1 ELSE 0 END) * 100.0 / COUNT(*), 1) as five_star_percent
-> FROM
->     restaurant r
-> LEFT JOIN
->     review rev ON r.restaurant_id = rev.restaurant_id
-> GROUP BY
->     r.restaurant_id, r.name;
+-----+
| restaurant_id | name          | total_ratings | avg_rating | five_star_percent |
+-----+
|      1 | Spice Palace   |            3 |      4.67 |             66.7 |
|      2 | Burger Heaven  |            2 |      3.50 |              0.0 |
|      3 | Pizza Paradise |            1 |      5.00 |            100.0 |
|      4 | Sushi Supreme  |            0 |      NULL |              0.0 |
|      5 | Veggie Delight |            0 |      NULL |              0.0 |
+-----+
5 rows in set (0.01 sec)

```