```sql
-- Create Customers table
CREATE TABLE Customers (
    customer_id INT PRIMARY KEY,
    name VARCHAR(50),
    email VARCHAR(50),
    phone VARCHAR(15),
    address VARCHAR(100),
    city VARCHAR(30)
);

-- Create Restaurants table
CREATE TABLE Restaurants (
    restaurant_id INT PRIMARY KEY,
    restaurant_name VARCHAR(50),
    cuisine_type VARCHAR(30),
    phone VARCHAR(15),
    address VARCHAR(100),
    city VARCHAR(30),
    rating DECIMAL(2,1)
);

-- Create Food_Items table
CREATE TABLE Food_Items (
    food_id INT PRIMARY KEY,
    food_name VARCHAR(50),
    restaurant_id INT,
    price DECIMAL(6,2),
    category VARCHAR(30),
    is_vegetarian VARCHAR(3),
    FOREIGN KEY (restaurant_id) REFERENCES Restaurants(restaurant_id)
);



-- Create Orders table
CREATE TABLE Orders (
    order_id INT PRIMARY KEY,
    customer_id INT,
    restaurant_id INT,
    food_id INT,
    quantity INT,
    total_amount DECIMAL(8,2),
    order_date DATE,
    delivery_time VARCHAR(20),
    status VARCHAR(20),
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id),
    FOREIGN KEY (restaurant_id) REFERENCES Restaurants(restaurant_id),
    FOREIGN KEY (food_id) REFERENCES Food_Items(food_id)
```

```sql
);

-- Insert Customers (8 customers)
INSERT INTO Customers VALUES
(1, 'Alice Johnson', 'alice@email.com', '555-2001', '123 Main St', 'New York'),
(2, 'Bob Smith', 'bob@email.com', '555-2002', '456 Oak Ave', 'Chicago'),
(3, 'Carol Davis', 'carol@email.com', '555-2003', '789 Pine Rd', 'Miami'),
(4, 'David Wilson', 'david@email.com', '555-2004', '321 Elm St', 'Seattle'),
(5, 'Emma Brown', 'emma@email.com', '555-2005', '654 Maple Dr', 'Boston'),
(6, 'Frank Miller', 'frank@email.com', '555-2006', '987 Cedar Ln', 'Denver'),
(7, 'Grace Lee', 'grace@email.com', '555-2007', '147 Birch Blvd', 'Austin'),
(8, 'Henry Taylor', 'henry@email.com', '555-2008', '258 Spruce Way', 'Portland');

-- Insert Restaurants (6 restaurants)
INSERT INTO Restaurants VALUES
(101, 'Pizza Palace', 'Italian', '555-3001', '100 Food St', 'New York', 4.5),
(102, 'Burger Barn', 'American', '555-3002', '200 Taste Ave', 'Chicago', 4.2),
(103, 'Spice Garden', 'Indian', '555-3003', '300 Flavor Rd', 'Miami', 4.7),
(104, 'Sushi Spot', 'Japanese', '555-3004', '400 Fresh St', 'Seattle', 4.6),
(105, 'Taco Town', 'Mexican', '555-3005', '500 Spicy Ave', 'Boston', 4.3),
(106, 'Noodle House', 'Chinese', '555-3006', '600 Wok Way', 'Denver', 4.4);

-- Insert Food_Items (12 food items)
INSERT INTO Food_Items VALUES
(201, 'Margherita Pizza', 101, 18.99, 'Main Course', 'Yes'),
(202, 'Pepperoni Pizza', 101, 21.99, 'Main Course', 'No'),
(203, 'Classic Burger', 102, 12.99, 'Main Course', 'No'),
(204, 'Veggie Burger', 102, 11.99, 'Main Course', 'Yes'),
(205, 'Chicken Curry', 103, 16.99, 'Main Course', 'No'),
(206, 'Dal Tadka', 103, 13.99, 'Main Course', 'Yes'),
(207, 'Salmon Roll', 104, 14.99, 'Main Course', 'No'),
(208, 'Veggie Roll', 104, 12.99, 'Main Course', 'Yes'),
(209, 'Beef Tacos', 105, 15.99, 'Main Course', 'No'),
(210, 'Bean Tacos', 105, 13.99, 'Main Course', 'Yes'),
(211, 'Chicken Noodles', 106, 14.99, 'Main Course', 'No'),
(212, 'Vegetable Noodles', 106, 12.99, 'Main Course', 'Yes');

-- Insert Orders (10 orders)
INSERT INTO Orders VALUES
(1001, 1, 101, 201, 2, 37.98, '2024-09-15', '30 minutes', 'Delivered'),
(1002, 2, 102, 203, 1, 12.99, '2024-09-16', '25 minutes', 'Delivered'),
```

```
(1003, 3, 103, 205, 1, 16.99, '2024-09-17', '35 minutes', 'Preparing'),
(1004, 4, 104, 207, 3, 44.97, '2024-09-18', '40 minutes', 'Delivered'),
(1005, 1, 105, 209, 2, 31.98, '2024-09-19', '28 minutes', 'On the way'),
(1006, 5, 106, 211, 1, 14.99, '2024-09-20', '32 minutes', 'Pending'),
(1007, 6, 101, 202, 1, 21.99, '2024-09-20', '30 minutes', 'Delivered'),
(1008, 7, 103, 206, 2, 27.98, '2024-09-21', '35 minutes', 'Preparing'),
(1009, 8, 104, 208, 1, 12.99, '2024-09-21', '40 minutes', 'On the way'),
(1010, 2, 105, 210, 3, 41.97, '2024-09-22', '28 minutes', 'Pending');


-- Query 1: Find all food orders from the last 7 days
SELECT Orders.order_id, Customers.name, Restaurants.restaurant_name,
Food_Items.food_name, Orders.order_date, Orders.status
FROM Orders
JOIN Customers ON Orders.customer_id = Customers.customer_id
JOIN Restaurants ON Orders.restaurant_id = Restaurants.restaurant_id
JOIN Food_Items ON Orders.food_id = Food_Items.food_id
WHERE Orders.order_date >= '2024-09-15'
ORDER BY Orders.order_date DESC;

-- Query 2: Top 3 most expensive food items by revenue
SELECT Food_Items.food_name, Restaurants.restaurant_name, SUM(Orders.quantity) AS
total_sold, SUM(Orders.total_amount) AS total_revenue
FROM Food_Items
JOIN Orders ON Food_Items.food_id = Orders.food_id
JOIN Restaurants ON Food_Items.restaurant_id = Restaurants.restaurant_id
GROUP BY Food_Items.food_id, Food_Items.food_name, Restaurants.restaurant_name
ORDER BY total_revenue DESC
LIMIT 3;

-- Query 3: Customers with pending or preparing orders
SELECT Customers.name, Customers.phone, Orders.order_id,
Restaurants.restaurant_name, Orders.status
FROM Customers
JOIN Orders ON Customers.customer_id = Orders.customer_id
JOIN Restaurants ON Orders.restaurant_id = Restaurants.restaurant_id
WHERE Orders.status IN ('Pending', 'Preparing')
ORDER BY Orders.order_date;




-- Query 4: All vegetarian food items available
SELECT Food_Items.food_name, Restaurants.restaurant_name, Food_Items.price,
Food_Items.category
FROM Food_Items
JOIN Restaurants ON Food_Items.restaurant_id = Restaurants.restaurant_id
WHERE Food_Items.is_vegetarian = 'Yes'
```

```
ORDER BY Food_Items.price ASC;

-- Query 5: Restaurant performance - total orders and revenue
SELECT Restaurants.restaurant_name, Restaurants.cuisine_type, COUNT(Orders.order_id)
AS total_orders, SUM(Orders.total_amount) AS total_revenue
FROM Restaurants
JOIN Orders ON Restaurants.restaurant_id = Orders.restaurant_id
GROUP BY Restaurants.restaurant_id, Restaurants.restaurant_name,
Restaurants.cuisine_type
ORDER BY total_revenue DESC;
```