PHASE-2 ASSIGNMENT

Title: Online Food Ordering System

Student Name: **NAMEERA TAHREEM K A**

Register Number: **31223U18023**

Institution Name: **ISLAMIAH WOMEN'S ARTS AND SCIENCE COLLEGE.**

Department: **BSC COMPUTER SCIENCE**

Date of Submission: **19.08.2025**

```sql
CREATE TABLE Customer1 (

    customer_id NUMBER(5) PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    email VARCHAR(100) UNIQUE NOT NULL,

    phone VARCHAR(15) NOT NULL,

    address VARCHAR(50)

);

INSERT INTO Customer1 VALUES

(1, 'Alice', 'alice@example.com', '9876543210', 'Delhi');

INSERT INTO Customer1 VALUES(2, 'Bob', 'bob@example.com', '9876543211', 'Mumbai');

INSERT INTO Customer1 VALUES(3, 'Charlie', 'charlie@example.com', '9876543212', 'Chennai');

INSERT INTO Customer1 VALUES(4, 'David', 'david@example.com', '9876543213', 'Kolkata');

INSERT INTO Customer1 VALUES(5, 'Eva', 'eva@example.com', '9876543214', 'Bangalore');

SELECT*FROM Customer1;

CREATE TABLE Restaurant (

    restaurant_id NUMBER(3) PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    location VARCHAR(100),

    contact_number VARCHAR(15)

);

CREATE TABLE FoodItem (

    item_id NUMBER(5) PRIMARY KEY,

    restaurant_id INT,

    item_name VARCHAR(100) NOT NULL,

    price DECIMAL(10, 2) NOT NULL,

    category VARCHAR(50),

    FOREIGN KEY (restaurant_id) REFERENCES Restaurant(restaurant_id)

);

INSERT INTO Restaurant VALUES(1, 'Spicy House', 'Delhi', '1345654321');

INSERT INTO Restaurant VALUES(2, 'Tandoori Tales', 'Mumbai', '2212345222');

INSERT INTO Restaurant VALUES(3, 'Veggie Delight', 'Chennai', '3987654233');
```

```sql
INSERT INTO FoodItem VALUES(1, 1, 'Paneer Tikka', 250.00, 'Veg');

INSERT INTO FoodItem VALUES(2, 1, 'Chicken Curry', 350.00, 'Non-Veg');

INSERT INTO FoodItem VALUES(3, 2, 'Veg Biryani', 200.00, 'Veg');

INSERT INTO FoodItem VALUES(4, 2, 'Mutton Rogan Josh', 450.00, 'Non-Veg');

INSERT INTO FoodItem VALUES(5, 3, 'Veg Sandwich', 150.00, 'Veg');

INSERT INTO FoodItem VALUES(6, 3, 'Fruit Salad', 100.00, 'Veg');

INSERT INTO FoodItem VALUES(7, 1, 'Butter Naan', 50.00, 'Veg');

INSERT INTO FoodItem VALUES(8, 2, 'Fish Fry', 300.00, 'Non-Veg');

SELECT F.item_name, F.price, R.name AS restaurant_name

FROM FoodItem F

JOIN Restaurant R ON F.restaurant_id = R.restaurant_id;


CREATE TABLE DeliveryAgent (

    agent_id INT PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    phone VARCHAR(15)

);

INSERT INTO DeliveryAgent VALUES(1, 'Rahul', '9000000001');

INSERT INTO DeliveryAgent VALUES(2, 'Sunita', '9000000002');

INSERT INTO DeliveryAgent VALUES(3, 'Karan', '9000000003');

CREATE TABLE Orders1 (order_id NUMBER(5) PRIMARY KEY,customer_id NUMBER(3) ,FOREIGN KEY
(customer_id) REFERENCES Customer1(customer_id),order_date DATE NOT NULL,total_amount
DECIMAL(10, 2),status VARCHAR(50));

INSERT INTO Orders1 VALUES(101, 1,'25-aug-2025', 600.00, 'Delivered');

INSERT INTO Orders1 VALUES(102, 2, '20-aug-2014', 400.00, 'Delivered');

INSERT INTO Orders1 VALUES(103, 1, '02-aug-2025', 800.00, 'Pending');

INSERT INTO Orders1 VALUES(104, 3, '10-aug-2025', 1200.00, 'Delivered');

INSERT INTO Orders1 VALUES(105, 4, '15-aug-2025', 300.00, 'Cancelled');

SELECT * FROM Orders1 WHERE customer_id = 1;

CREATE TABLE OrderDetails1 (

    order_id NUMBER(5),

    item_id NUMBER(5),
```

```sql
 quantity NUMBER(5),

    PRIMARY KEY (order_id, item_id),

    FOREIGN KEY (order_id) REFERENCES Orders1(order_id),

    FOREIGN KEY (item_id) REFERENCES FoodItem(item_id)

); INSERT INTO OrderDetails1 VALUES(101, 7, 4); INSERT INTO OrderDetails1 VALUES(102, 3, 2);

INSERT INTO OrderDetails1 VALUES(103, 4, 2); INSERT INTO OrderDetails1 VALUES(103, 8, 1); INSERT

INTO OrderDetails1 VALUES(104, 2, 2); INSERT INTO OrderDetails1 VALUES(105, 5, 2); INSERT INTO

OrderDetails1 VALUES(101, 2, 1); CREATE TABLE Review(review_id NUMBER(5) PRIMARY

KEY,customer_id NUMBER(5),FOREIGN KEY
(customer_id) REFERENCES Customer1(customer_id),
```

```sql
    restaurant_id NUMBER(3),FOREIGN KEY (restaurant_id)REFERENCES Restaurant(restaurant_id),

rating NUMBER(6) CHECK (rating BETWEEN 1 AND 5),comments VARCHAR(30));

INSERT INTO Review VALUES(1, 1, 1, 5, 'Great food and delivery!');

INSERT INTO Review VALUES(2, 2, 2, 4, 'Tasty biryani.');

INSERT INTO Review VALUES(3, 3, 3, 3, 'Good, but can improve.');

CREATE TABLE Offer (offer_id NUMBER(3) PRIMARY KEY,restaurant_id NUMBER(3),FOREIGN KEY
(restaurant_id) REFERENCES Restaurant(restaurant_id),

item_id NUMBER(5),FOREIGN KEY (item_id) REFERENCES FoodItem(item_id),discount_percent
DECIMAL(5,2),
valid_till DATE

);

INSERT INTO Offer VALUES(1, 1, 1, 10.00, '20-AUG-2025');

INSERT INTO Offer VALUES(2, 2, 4, 15.00, '15-AUG-2025');

SELECT R.name AS restaurant_name, COUNT(F.item_id) AS total_items

FROM Restaurant R

JOIN FoodItem F ON R.restaurant_id = F.restaurant_id

GROUP BY R.name;

SELECT * FROM Orders1 WHERE total_amount > 500;
```

SELECT agent_id,name FROM DeliveryAgent WHERE agent_id IN(SELECT agent_id FROM Orders1 GROUP BY agent_id HAVING COUNT(*)>3);

SELECT * FROM Review WHERE restaurant_id = 1;

SELECT F.item_name, F.price, R.name AS restaurant_name

FROM FoodItem F

JOIN Restaurant R ON F.restaurant_id = R.restaurant_id

ORDER BY F.price DESC

FETCH FIRST 1 ROWS ONLY;

SELECT * FROM Offer

WHERE valid_till >= '20-AUG-2025';

SELECT SUM(total_amount) AS total_revenue

FROM Orders

WHERE status = 'Delivered';

1. Display all customers.

CUSTOMER_ID	NAME	EMAIL	PHONE	ADDRESS
1	Alice	alice@example.com	9876543210	Delhi
2	Bob	bob@example.com	9876543211	Mumbai
5	Eva	eva@example.com	9876543214	Bangalore
3	Charlie	charlie@example.com	9876543212	Chennai
4	David	david@example.com	9876543213	Kolkata

2. List all food items along with their restaurant names.

ITEM_NAME	PRICE	RESTAURANT_NAME
Chicken Curry	350	Spicy House
Veg Biryani	200	Tandoori Tales
Veg Sandwich	150	Veggie Delight
Fish Fry	300	Tandoori Tales
Paneer Tikka	250	Spicy House
Mutton Rogan Josh	450	Tandoori Tales
Fruit Salad	100	Veggie Delight
Butter Naan	50	Spicy House

3. Show all orders placed by a specific customer.

ORDER_ID	CUSTOMER_ID	ORDER_DATE	TOTAL_AMOUNT	STATUS
103	1	02-AUG-25	800	Pending
101	1	25-AUG-25	600	Delivered

4. Find the total number of food items available in each restaurant.

RESTAURANT_NAME	TOTAL_ITEMS
Spicy House	3
Tandoori Tales	3
Veggie Delight	2

5. Display orders with their total amount greater than ₹500.

ORDER_ID	CUSTOMER_ID	ORDER_DATE	TOTAL_AMOUNT	STATUS
103	1	02-AUG-25	800	Pending
104	3	10-AUG-25	1200	Delivered
101	1	25-AUG-25	600	Delivered

6. List delivery agents who have delivered more than 3 orders.

AGENT_ID	NAME
2	Sunita
1	Rahul
3	Karan

7. Show all reviews for a particularrestaurant.

REVIEW_ID	CUSTOMER_ID	RESTAURANT_ID	RATING	COMMENTS
1	1	1	5	Great food and delivery!

8. Find the most expensive food item and its restaurant.

ITEM_NAME	PRICE	RESTAURANT_NAME
Mutton Rogan Josh	450	Tandoori Tales

9. Display all offers currently valid today.

OFFER_ID	RESTAURANT_ID	ITEM_ID	DISCOUNT_PERCENT	VALID_TILL
1	1	1	10	20-AUG-25

10.Calculate the total revenue generated from all orders.

TOTAL_REVENUE
-