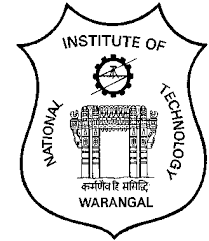
****

**PROBLEM STATEMENT:**

**Online Food Delivery Database Management System** aims to provide a comprehensive platform that enables customers to order their favourite food from various restaurants and have it delivered to their location. It helps to eliminate the difficulty of conventional queuing mechanisms. This system enhances the process of food order by a particular customer, from the ordering of food, looking at the menu being provided online, to the payment of the money as they wish, be it an online payment or the payment of cash, after the food is delivered.

It efficiently manages and organizes the details such as customer profiles, restaurant menus, order details, delivery logistics, and payment transactions, making the process for customers as easy as that being done in person. The customer is also provided with a facility of giving feedback regarding the food item received and the hotel as well, along with the rating. The database administrator can insert and delete data like restaurant details, delivery person details, food items, etc.,

**ASSUMPTIONS:**

* A customer with a unique customer id can order any number of different food items with a single order id.
* Several orders with different order ids can also be placed by a customer.
* Multiple order ids can have a single customer id.
* Each order id has a unique transaction id.
* Each type of food ordered has a unique id that links to give information on the name of the food item and its price per unit value.
* Each restaurant has its unique id to give information regarding its name, location and rating.
* Each delivery partner has a unique id to give information regarding their name, contact number and rating.
* Each customer has a unique id to give information regarding their name, contact number and address.
* Each customer can only order from one restaurant.

**TABLES:**

**1. CUSTOMER**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| customer\_id | INT (5) | PRIMARY KEY |
| customer\_name | VARCHAR (30) | NOT NULL |
| customer\_contact\_number | VARCHAR (10) | NOT NULL |
| customer\_address | VARCHAR (40) | NOT NULL |

**2. RESTAURANT**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| res\_id | INT (3) | PRIMARY KEY |
| res\_name | VARCHAR (20) | NOT NULL |
| res\_location | VARCHAR (20) | NOT NULL |
| res\_rating | DECIMAL (2,1) | NOT NULL |

**3. ORDER DETAIL**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| order\_id | INT(4) | PRIMARY KEY |
| customer \_id | INT (5) | NOT NULL, FOREIGN KEY |
| res\_id | INT(3) | NOT NULL, FOREIGN KEY |
| deliverp\_id | INT(4) | NOT NULL, FOREIGN KEY |
| order\_status | VARCHAR(10) | NOT NULL |
| ordered\_time | TIMESTAMP | NOT NULL |
| delivered\_time | TIMESTAMP | NOT NULL |

**4. DELIVERY PARTNER**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| deliveryp\_id | INT (4) | PRIMARY KEY |
| deliveryp\_name | VARCHAR (30) | NOT NULL |
| deliveryp\_contact\_number | VARCHAR (10) | NOT NULL |
| deliveryp\_avg\_rating | DECIMAL (2,1) | NOT NULL |

**5. FOODS**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| food\_id | INT (7) | PRIMARY KEY |
| food\_name | VARCHAR (20) | NOT NULL |
| price\_per\_unit | DECIMAL (5,2) | NOT NULL |

**6. ORDER FOOD**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| order\_id | INT (4) | PRIMARY KEY, FOREIGN KEY |
| food\_id | INT (7) | PRIMARY KEY, FOREIGN KEY |
| quantity | INT (3) | NOT NULL |

**7. PAYMENT TABLE**

| **ATTRIBUTES** | **DATA TYPES** | **CONSTRAINTS** |
| --- | --- | --- |
| payment\_id | INT (8) | PRIMARY KEY |
| order\_id | INT (4) | NOT NULL, FOREIGN KEY |
| payment\_type | VARCHAR (20) | NOT NULL |
| payment\_status | VARCHAR (10) | NOT NULL |

**FUNCTIONAL DEPENDENCIES AND PRIMARY KEY:**

**1. CUSTOMER:**

customer\_id -> { customer\_name,contact\_number,address}

As all the fields are depending on customer\_id, (customer\_id) +->R.

Hence, **customer\_id** is a primary key.

**2. RESTAURANT:**

res\_id -> {res\_name,res\_location,res\_rating}

As all the fields are depending on restaurant\_id, (restaurant\_id) +->R.

Hence, **restaurant\_id** is a primary key.

**3. FOODS:**

food\_id -> {food\_name,price\_per\_unit}

As all the fields are depending on food\_id, (food\_id) +->R.

Hence, **food\_id** is a primary key.

**4. DELIVERY PARTNER:**

deliveryp\_id

->{deliveryp\_name,deliveryp\_contact\_number,deliveryp\_avg\_rating}

As all the fields are depending on deliveryp\_id, (deliveryp\_id) +->R.

Hence, **deliveryp\_id** is a primary key.

**5. ORDER DETAIL:**

order\_id -> {customer\_id,res\_id,deliverp\_id,order\_status,order\_time,delivered\_time}

As all the fields are depending on order\_id, (order\_id) +->R.

Hence, **order\_id** is a primary key.

Also, customer\_id,res\_id, and deliveryp\_id are foreign keys from customer, restaurant, and delivery\_partner tables respectively.

**6. ORDER FOOD:**

{order\_id + food\_id} -> quantity

As all the fields are depending on (order\_id + food\_id), (order\_id + food\_id) +->R.

The combination of order\_id and food\_id acts as a composite key to uniquely identify each row.

Also, **order\_id** and **food\_id** are foreign keys from the order\_detail and foods table respectively.

**7. PAYMENT TABLE:**

payment\_id -> { payment\_type,payment\_status,order\_id}

As all the fields are depending on payment\_id, (payment\_id) +->R.

Hence, **payment\_id** is a primary key.

Also, order\_id is a foreign key from the order\_detail table.

**NORMALISATION:**

**1. CUSTOMER:**

Primary key: customer\_id

All attributes depend on the customer\_id, hence the table is in 2NF.

All attributes depend directly on customer\_id hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**2. RESTAURANT:**

Primary key: res\_id

All attributes depend on the res\_id, hence the table is in 2NF.

All attributes depend directly on res\_id hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**3. FOODS:**

Primary key: food\_id

All attributes depend on the food\_id, hence the table is in 2NF.

All attributes depend directly on food\_id hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**4. DELIVERY PARTNER:**

Primary key: deliveryp\_id

All attributes depend on the deliveryp\_id, hence the table is in 2NF.

All attributes depend directly on deliveryp\_id hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**5. ORDER DETAIL:**

Primary key: order\_id

All attributes depend on the order\_id, hence the table is in 2NF.

All attributes depend directly on order\_id hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**6. ORDER FOOD:**

Primary key: order\_id, food\_id

All attributes depend on the order\_id and food\_id, hence the table is in 2NF.

All attributes depend directly on order\_id and food\_id, hence the table is in 3NF.

All dependencies are from Candidate Key. Therefore, the table is in BCNF.

**7. PAYMENT TABLE:**

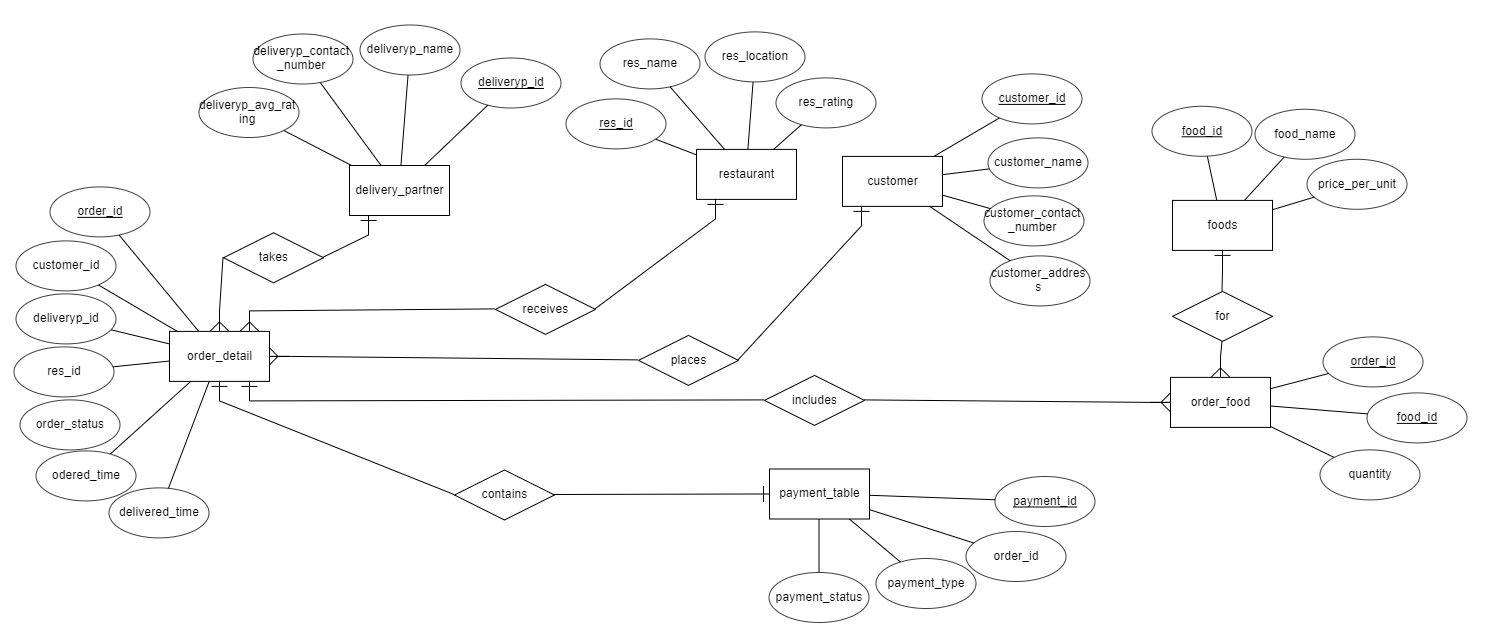
Primary key: payment\_id

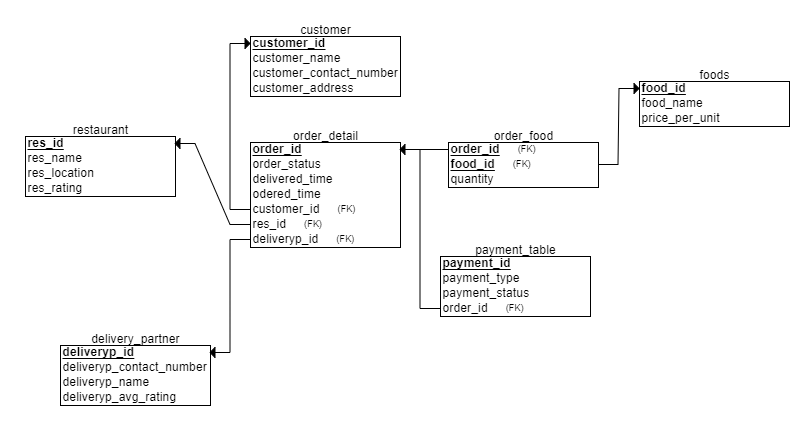
All attributes depend on the payment\_id, hence the table is in 2NF.

All attributes depend directly on payment\_id hence the table is in 3NF.

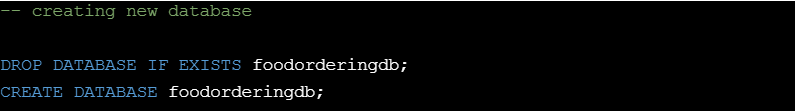
All dependencies are from Candidate Key. Therefore, the table is in BCNF.

Entity Relationship (ER) Diagram:

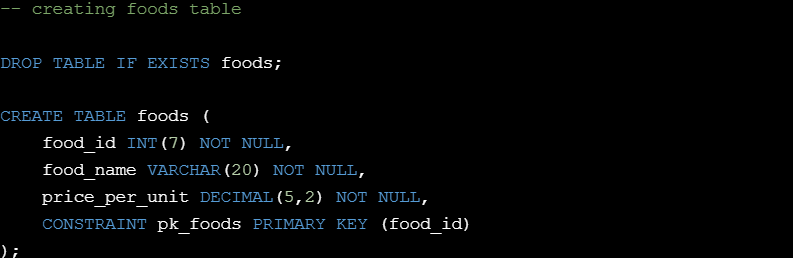
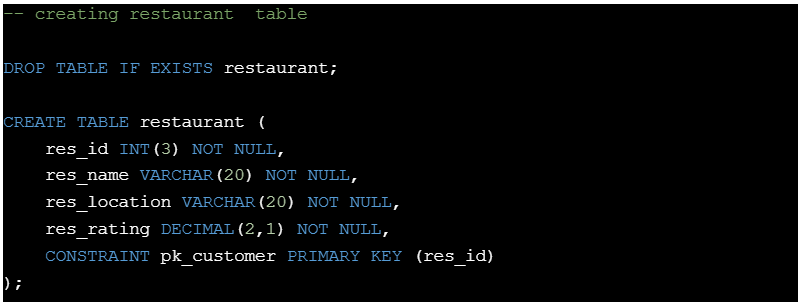
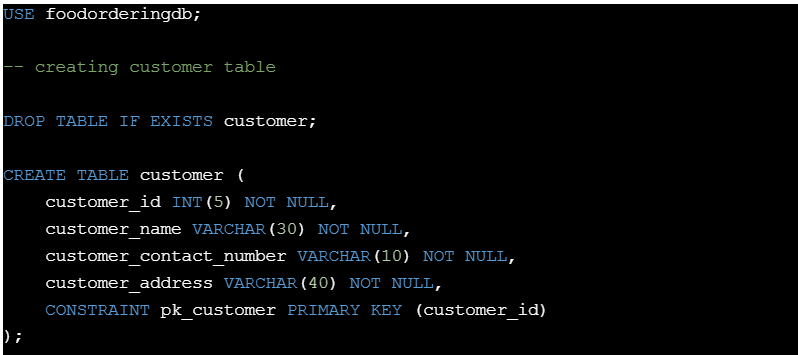
****

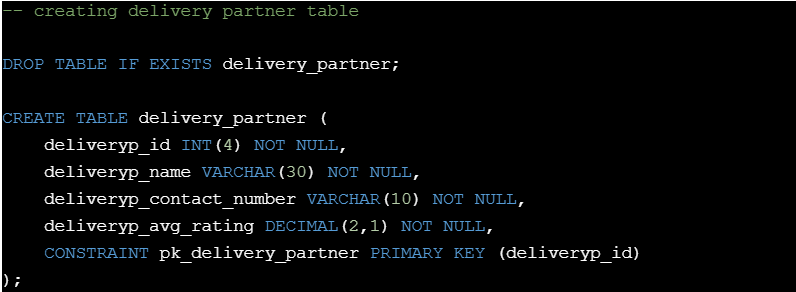
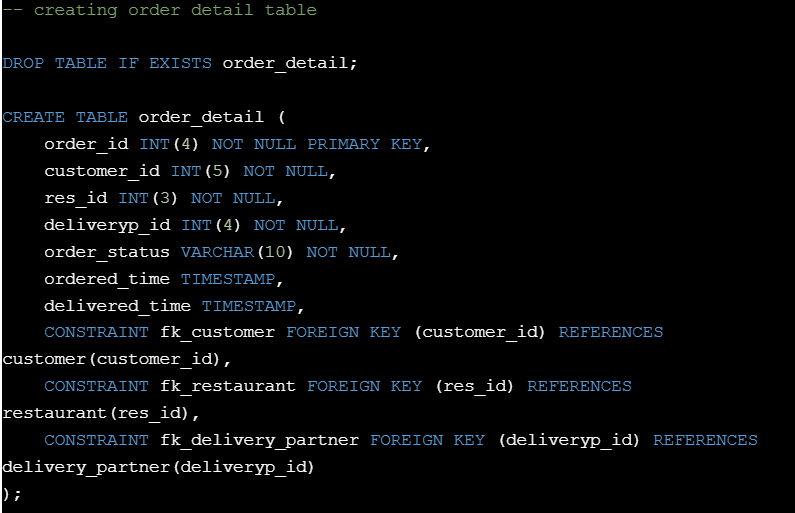
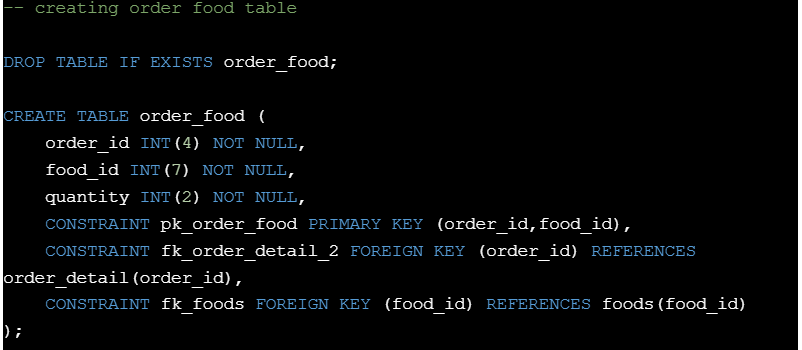
****Relational Schema:

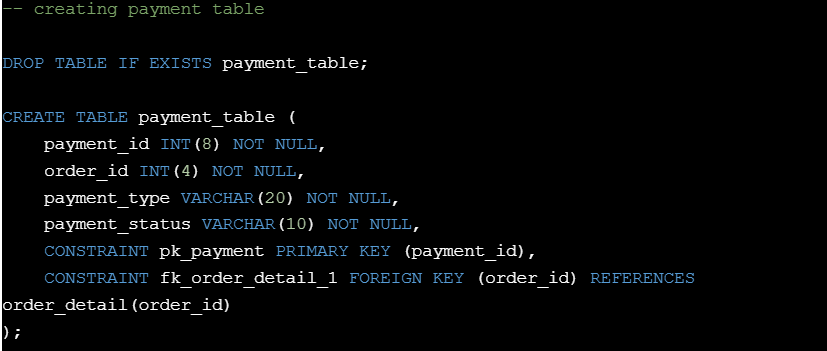
**SQL CODE:**

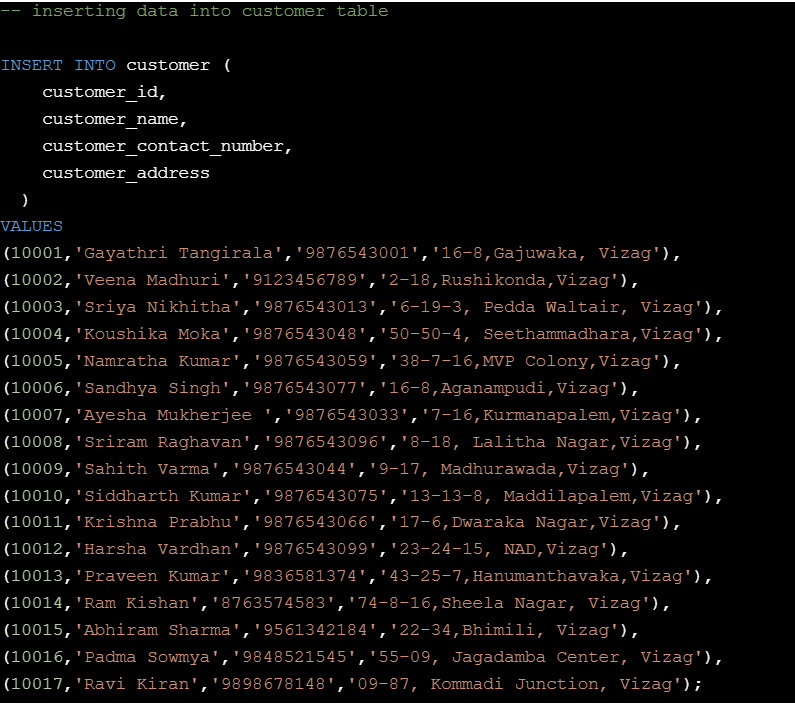
Creating Database:

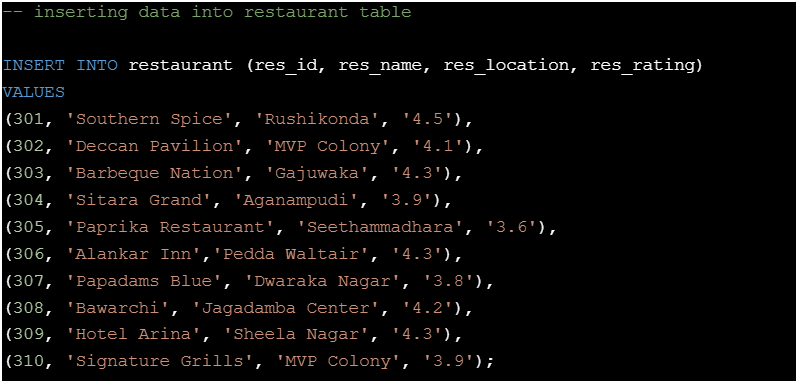
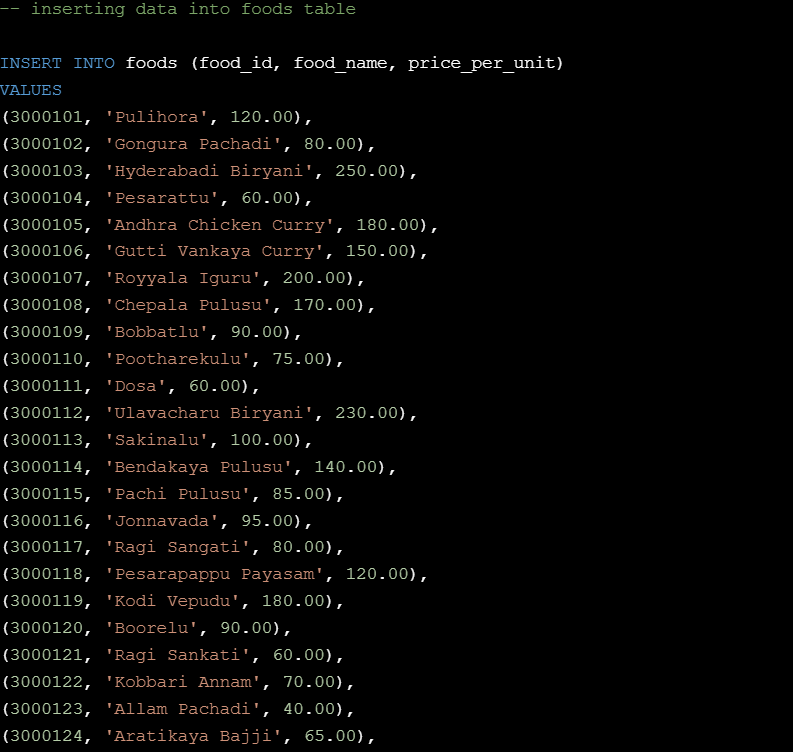
foodorderingdb;

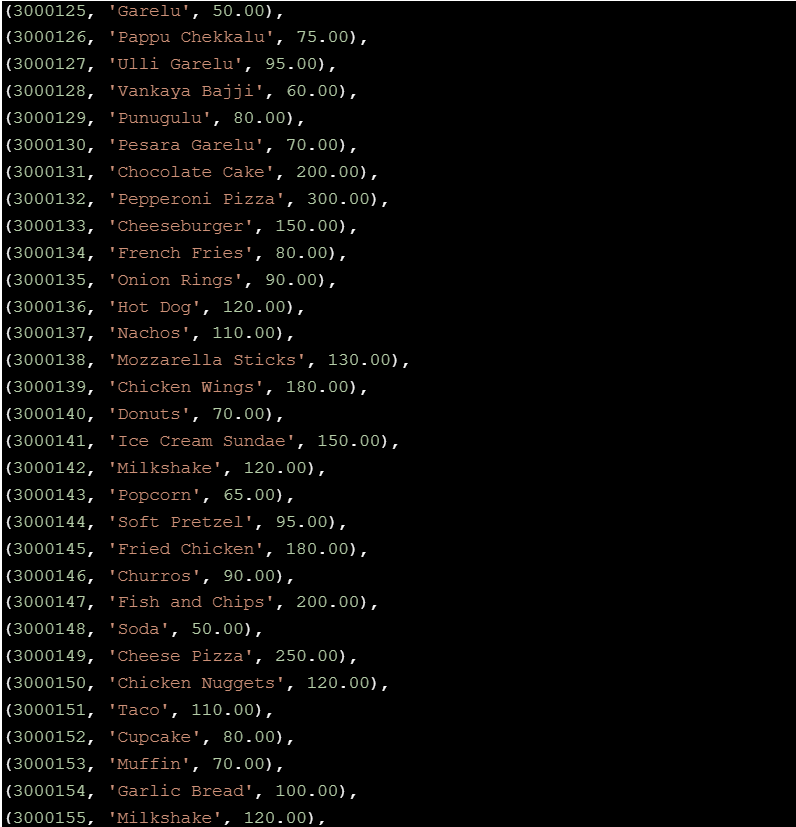
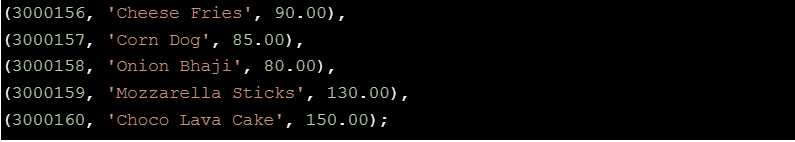
****Creating Tables:

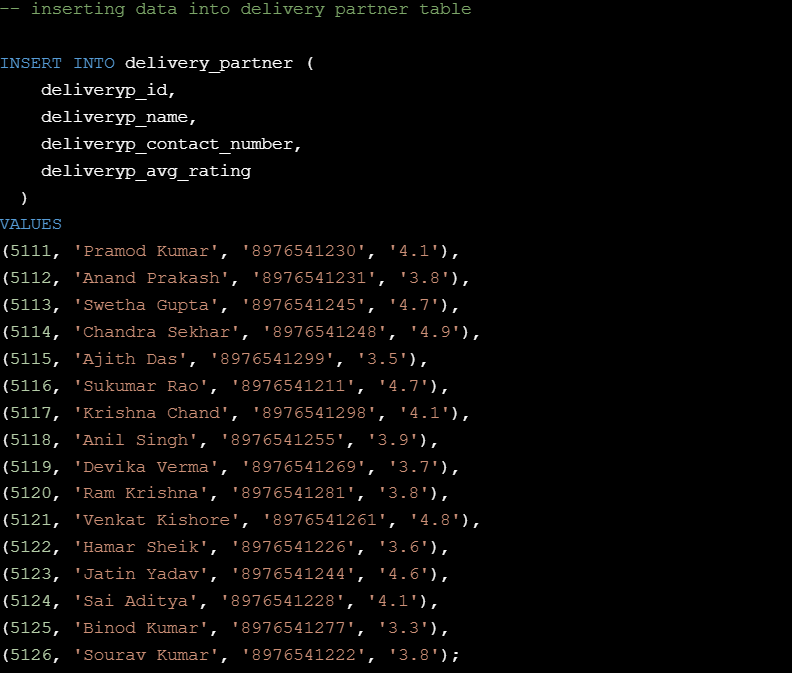
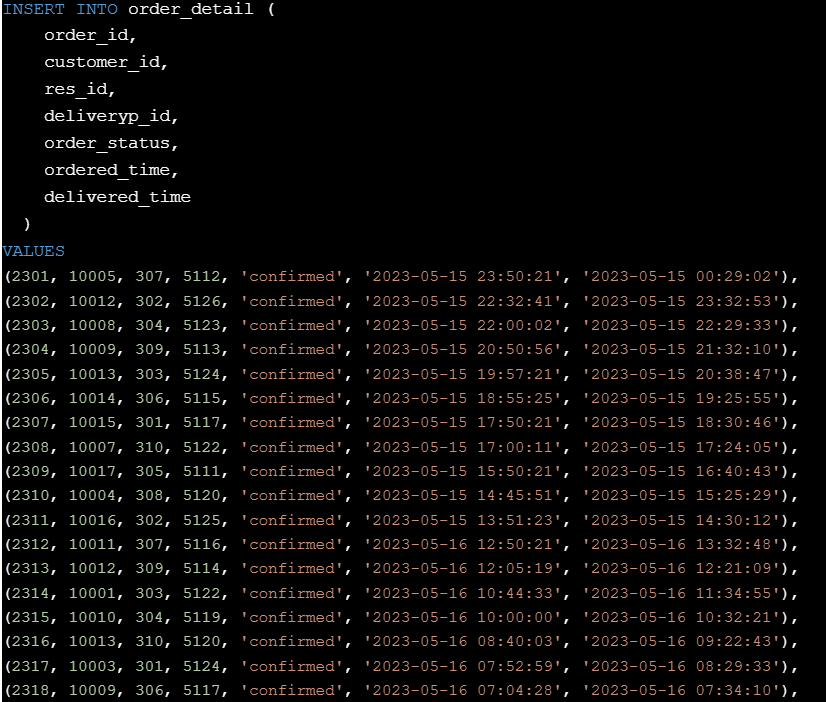


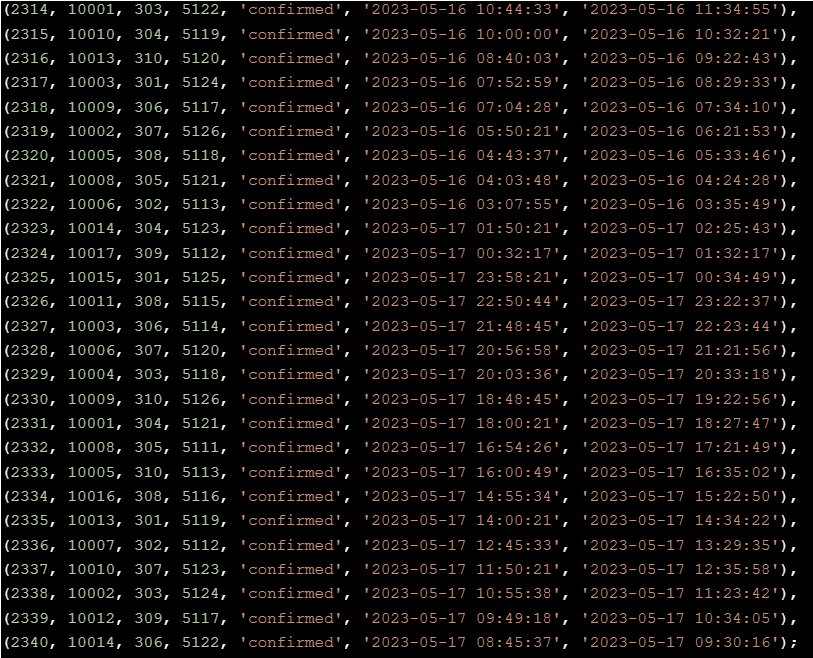
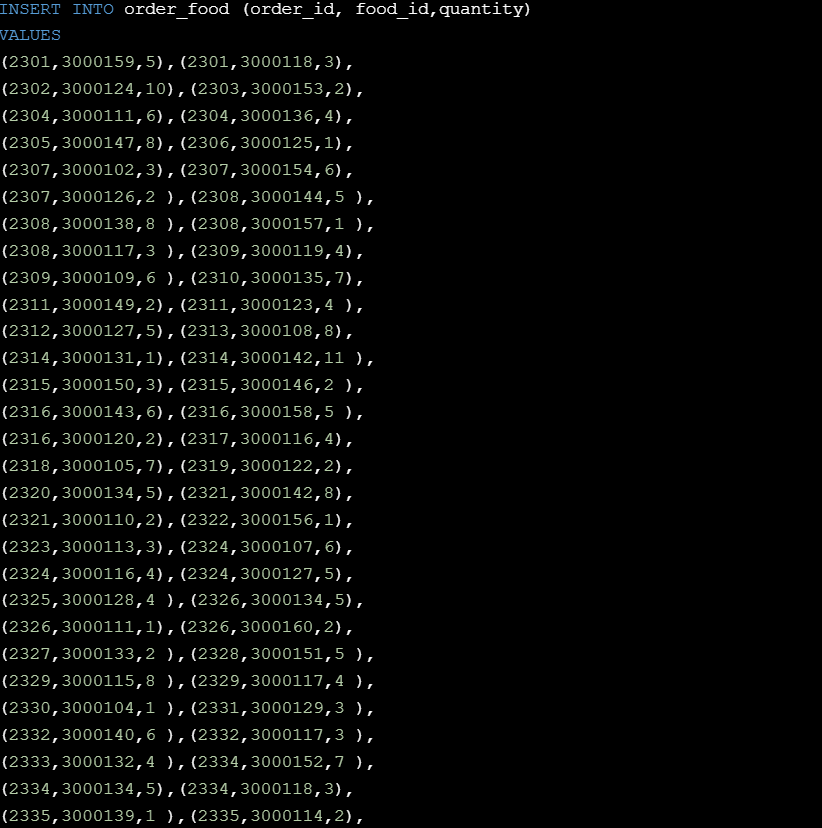


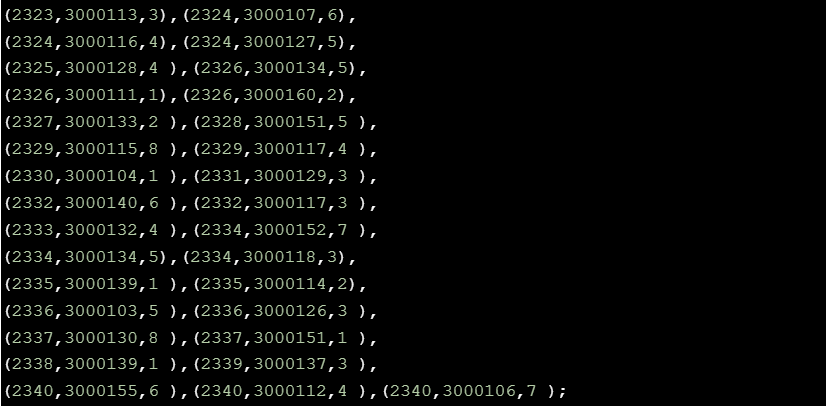
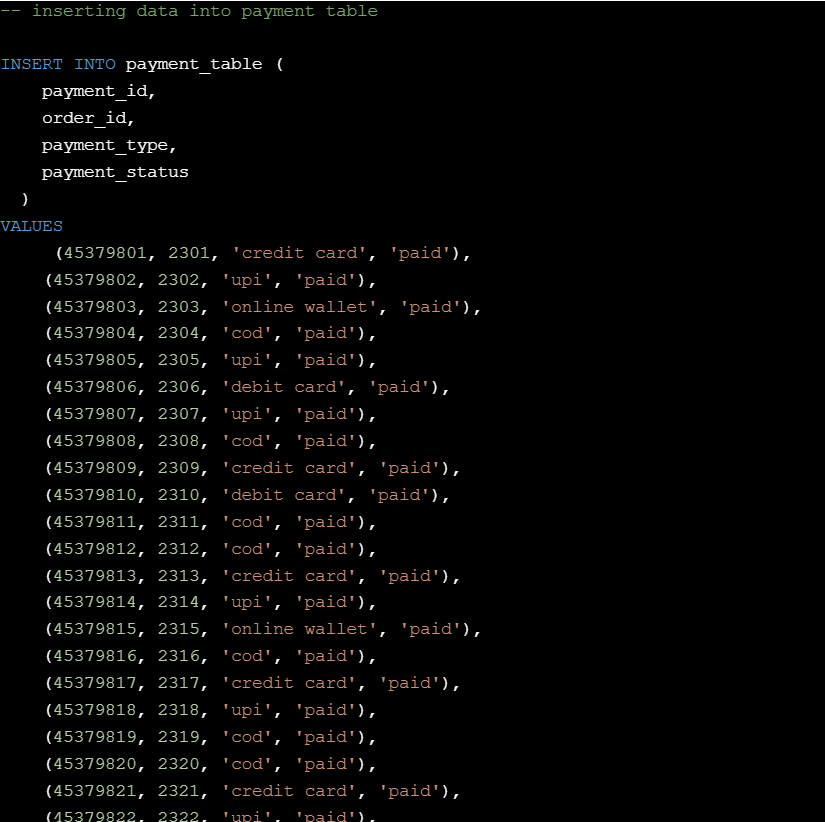
Inserting Data into Tables:

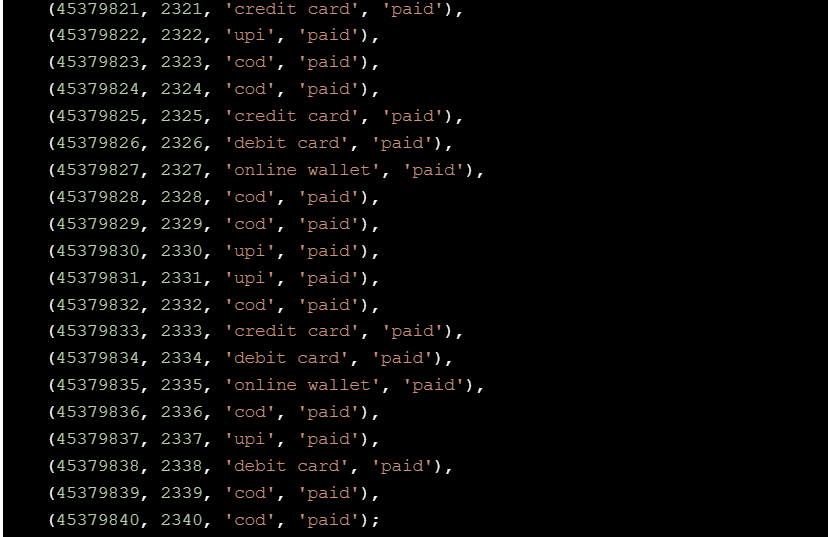




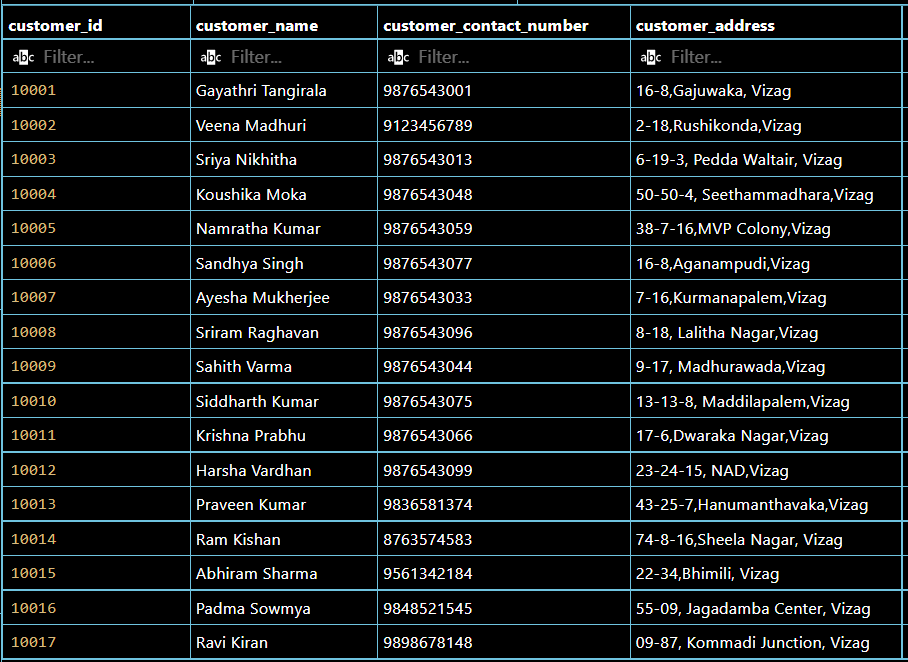
-

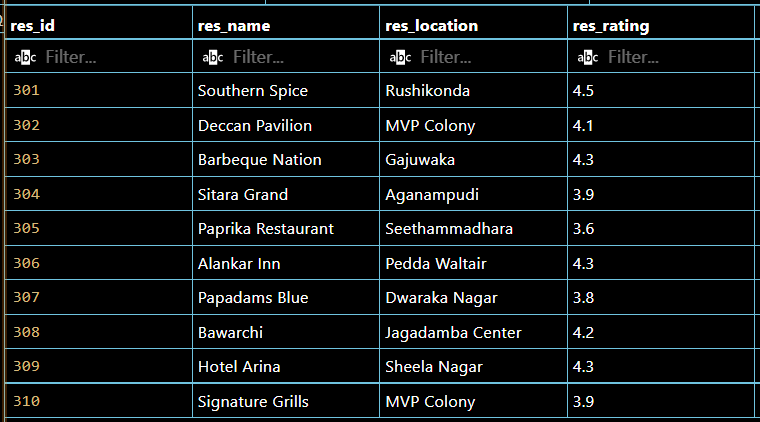






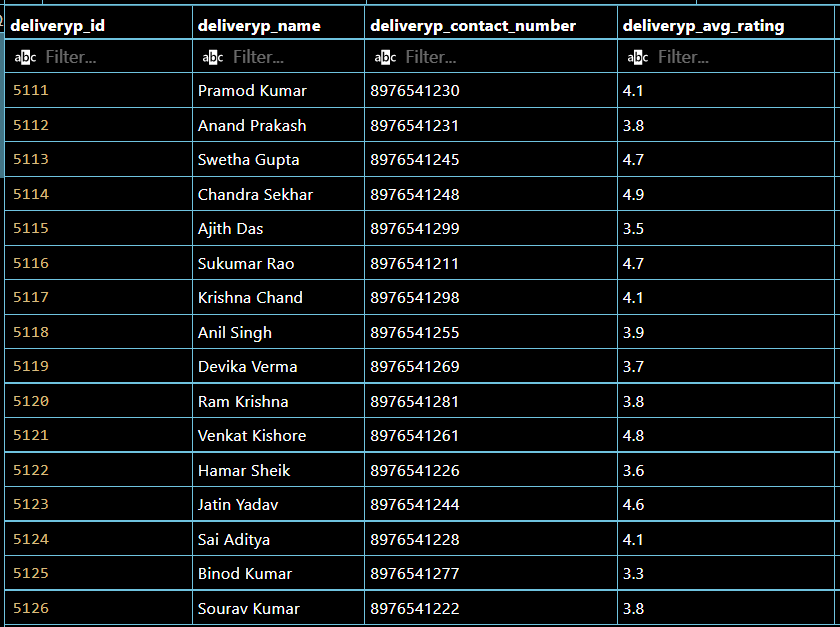
**TABLES:**

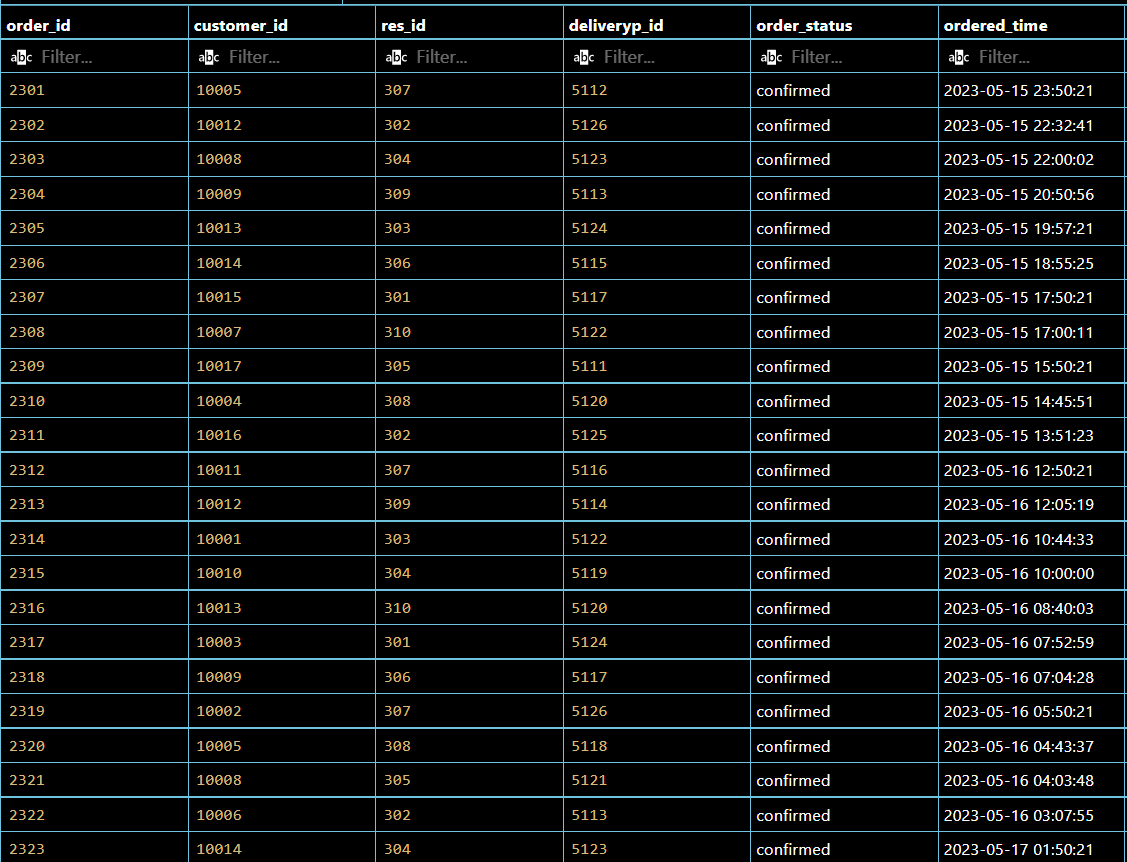
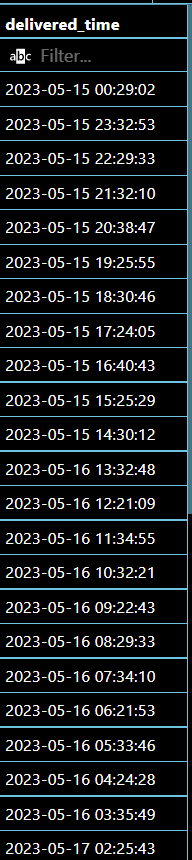
Customer Table:

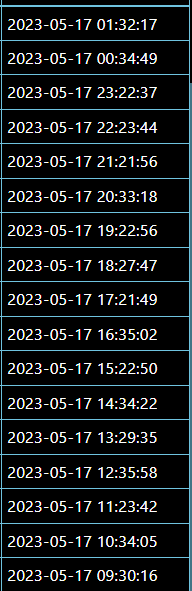
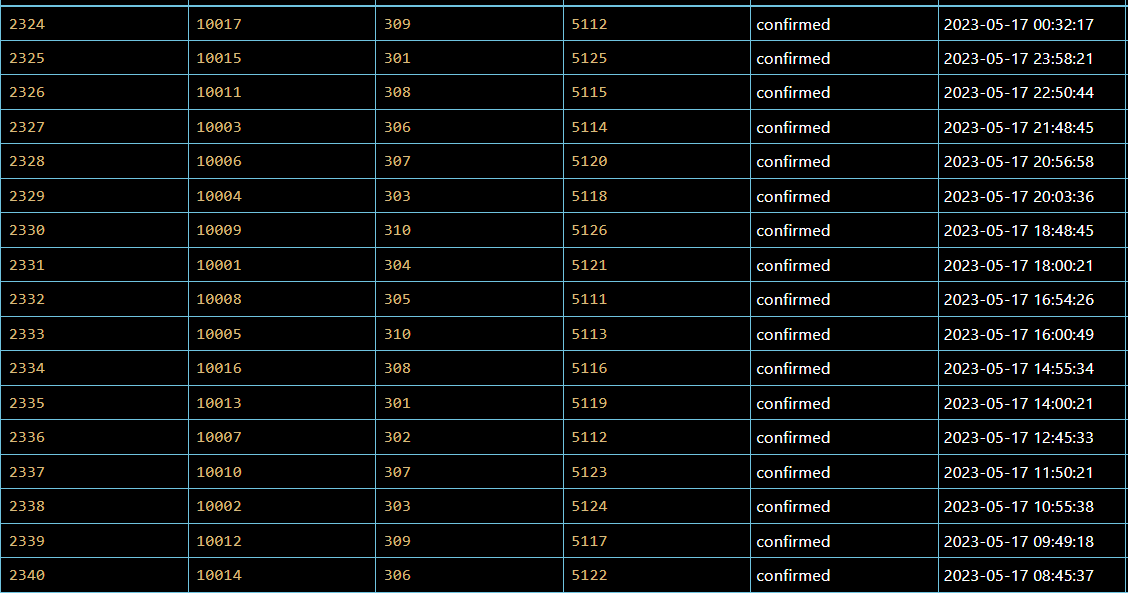
****Restaurant Table:

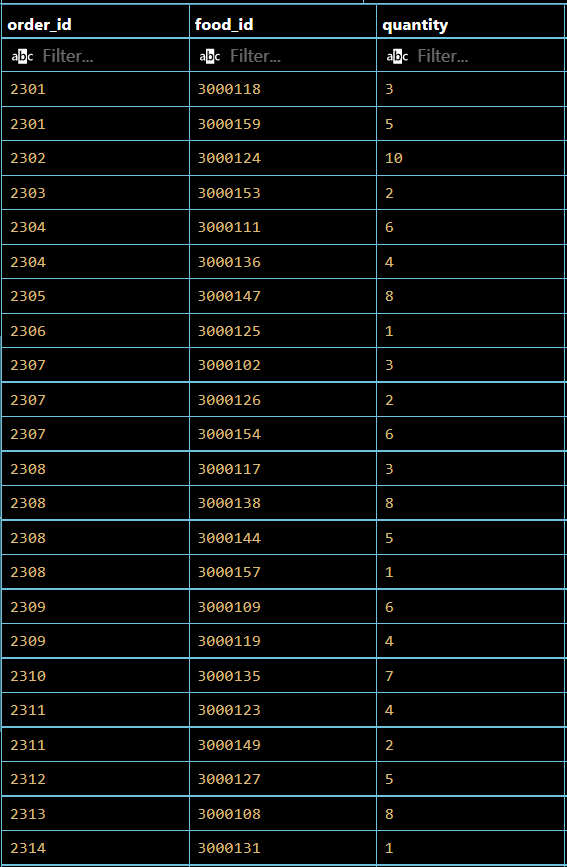
Foods Table:

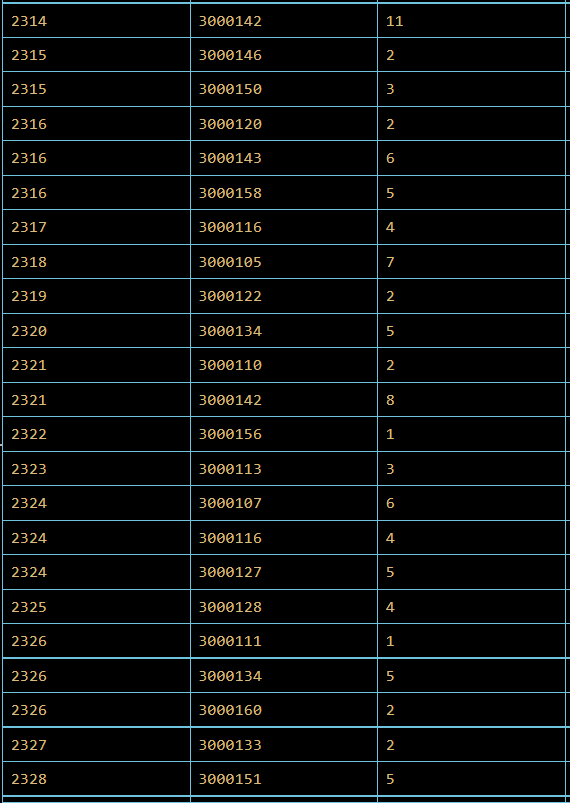
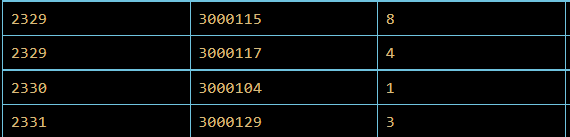
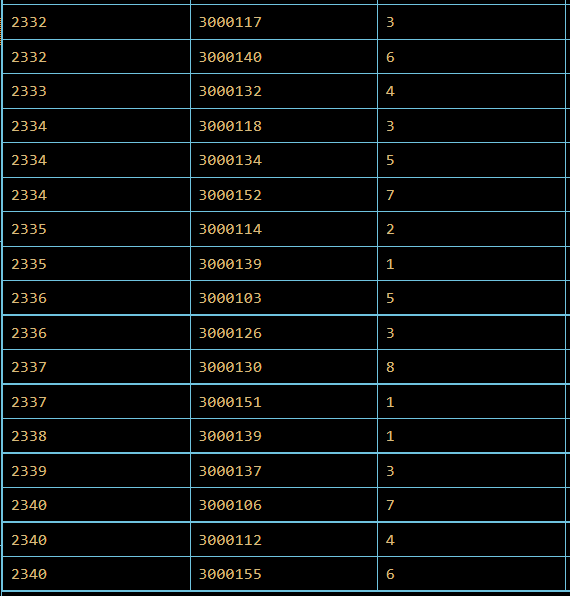


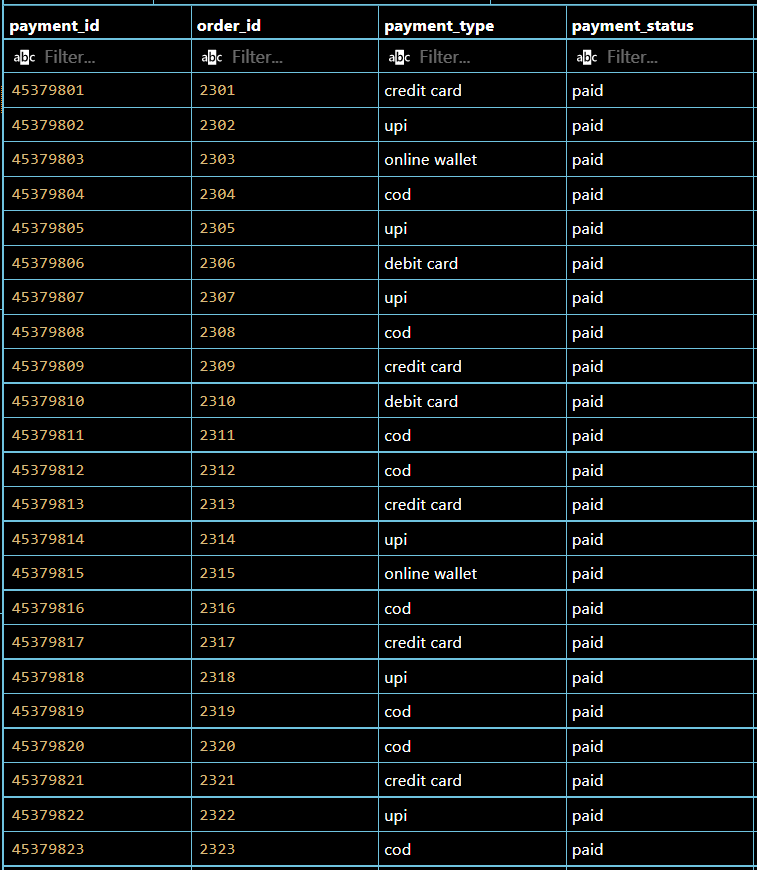
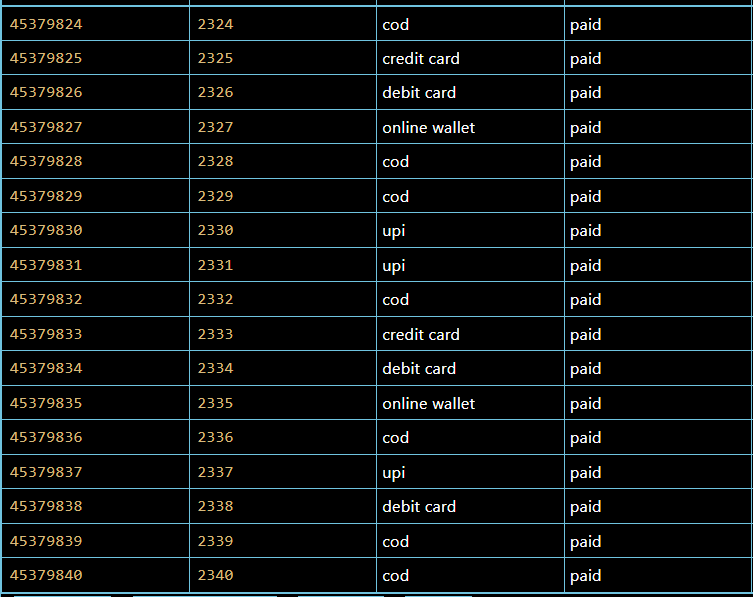
Delivery Partner Table:

Order Details Table :

****

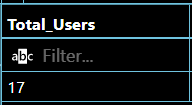
****Order Food Table :

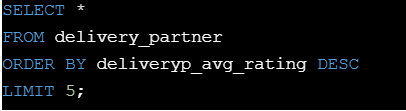
****

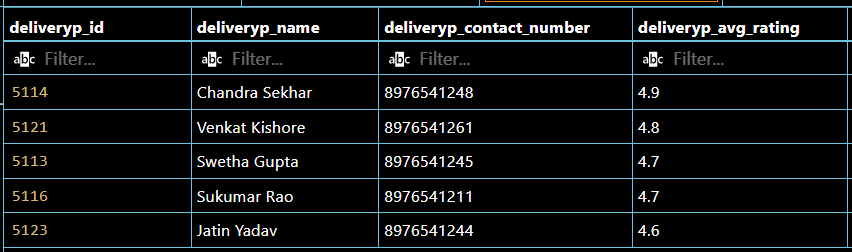
Payment Table:

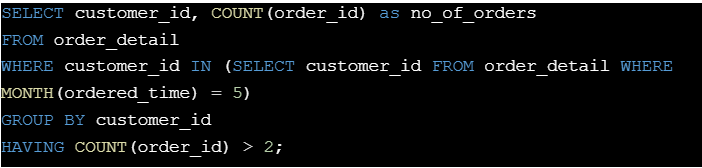
**QUERIES :**

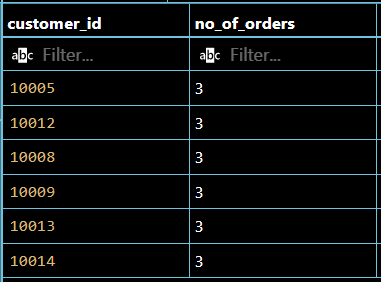
* Write a SQL query to find the total number of users :

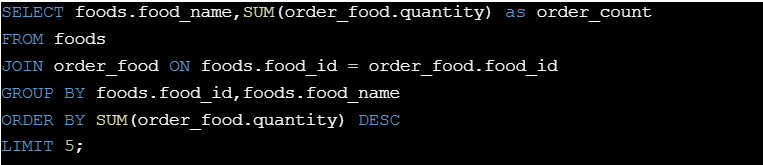
Output:

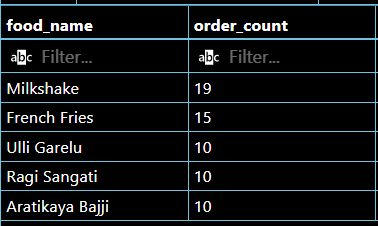
* Write a SQL query to find the top 5 delivery partners on the basis of ratings given by customers :

Output:

* Write a SQL query to find the list of users who made more than 2 orders in a May month :

Output:

* Write an SQL query to find the top 5 food items based on the quantity placed :

Output: