VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Restaurant Management System

Project Report for
ITE1003- Database Management System

by

YUVRAJ KUMAR 18BIT0276

ROUNAK BASAK: 18BIT0277

ARYAN SRIVASTAVA 18BIT0301

ISHAN AGGARWAL 18BIT0310

Submitted to

Prof. Bimal Kumar Ray

School of Information Technology and Engineering

Vellore Institute of Technology, Vellore-632014

Jul-Nov, 2019

FALL 2019-20

ABSTRACT

We have created a Database for a Restaurant as a part of our project for ITE1003 – Database Management System. It has been developed by using Oracle SQLPLUS software. In our database we have considered all the major functionalities that are required in a restaurant. An organized system in any restaurant makes the job a lot easier to manage everything properly. There are lots of different kinds of data to consider here. The various food details have to be stored. We should also have a record of all the customers who come to the restaurant. We need to maintain a proper ordering data too to keep track of the various food items ordered by the customer. The feedback given the customer and table number where they sat may also be needed for the future reference. Finance is one of the most important components here which has to be maintained really well too. Then comes the people who work in the restaurant and provide service to their customers. All of their details should also be kept properly. The entire project is not only about just storing the data. The process of retrieving the data also should not be troublesome. The users should be able to access it easily and the process should be fast too. So, all the related data should be interlinked which will help in these cases. Many times, the users might also want to insert new data, update or delete them too. Those cases also we have considered. Even after these modifications the database should work perfectly. Finally in a nutshell, our work is useful for storing all the data in an organized and efficient manner, saving valuable time and reduces the huge paper work.

DATA REQUIREMENTS & CONSTRAINTS

The Restaurant Management System must have the following data requirements and constraints:

Customers who come to the restaurant are identified by their customer IDs and their names, ages, phone numbers and addresses are recorded. Customer name should always be present. Customers give orders. Each Order is identified by order ID and has foodordered, bill no. and date. Bill no. and table no. should be unique. Maximum number of food items that can be ordered is 20. Payment has to be done for order. Payment is identified by payment ID, it also contains mode of payment (Cash, Debit card, Credit card and UPI), total amount, amount received, amount returned and date. Total amount should be equal to amount received – amount returned and should always be greater than zero. There are three kinds of staff members in the restaurant – Manager, Chef and Waiter. Every staff member is identified by staff ID and has a name, date of birth, contact number, address and salary. Staff name and contact should be present for every staff. Waiters' salary should be greater than ₹100. Chef's salary should be greater than ₹500. Manager's salary should be greater than ₹1000. Chef cooks the food and waiter takes and serves the orders. Order consists of varieties of food. Every food is identified by its food ID, it also has a name, description and price. Price should be greater than ₹100 and lesser than ₹1000. Food name should be present for every food item. Customers are at tables. Tables are identified by table number and has a number of seats. Number of seats should be more than 4. Customer gives feedback. Each feedback is identified by a Feedback ID, and contains ratings (1-5) and comments.

FUNCTIONAL REQUIREMENTS

Insertion of Data

1. Details of customers are stored.
2. Details of the orders placed are stored.
3. Details of the dishes are stored.
4. Details of the tables are stored.
5. Details of the payments done are stored.
6. Feedback given by customers is also stored.
7. Details of the staff (manager, chef, waiter) of the restaurant are stored.
8. Details of which cooked which food are stored
9. Details of which waiter served which order are stored
10. Details of which dishes are ordered in a particular order are stored

Removal of old data

1. A customer decides to delete his order at a particular date. So by his id and date of order, the staff id , food id and order id should be deleted.
2. A customer decides to delete a food item from his order. So its food id needs to be deleted.
3. An order has been placed to cook which gets rejected at last moment. So its food id for cook needs to be deleted.

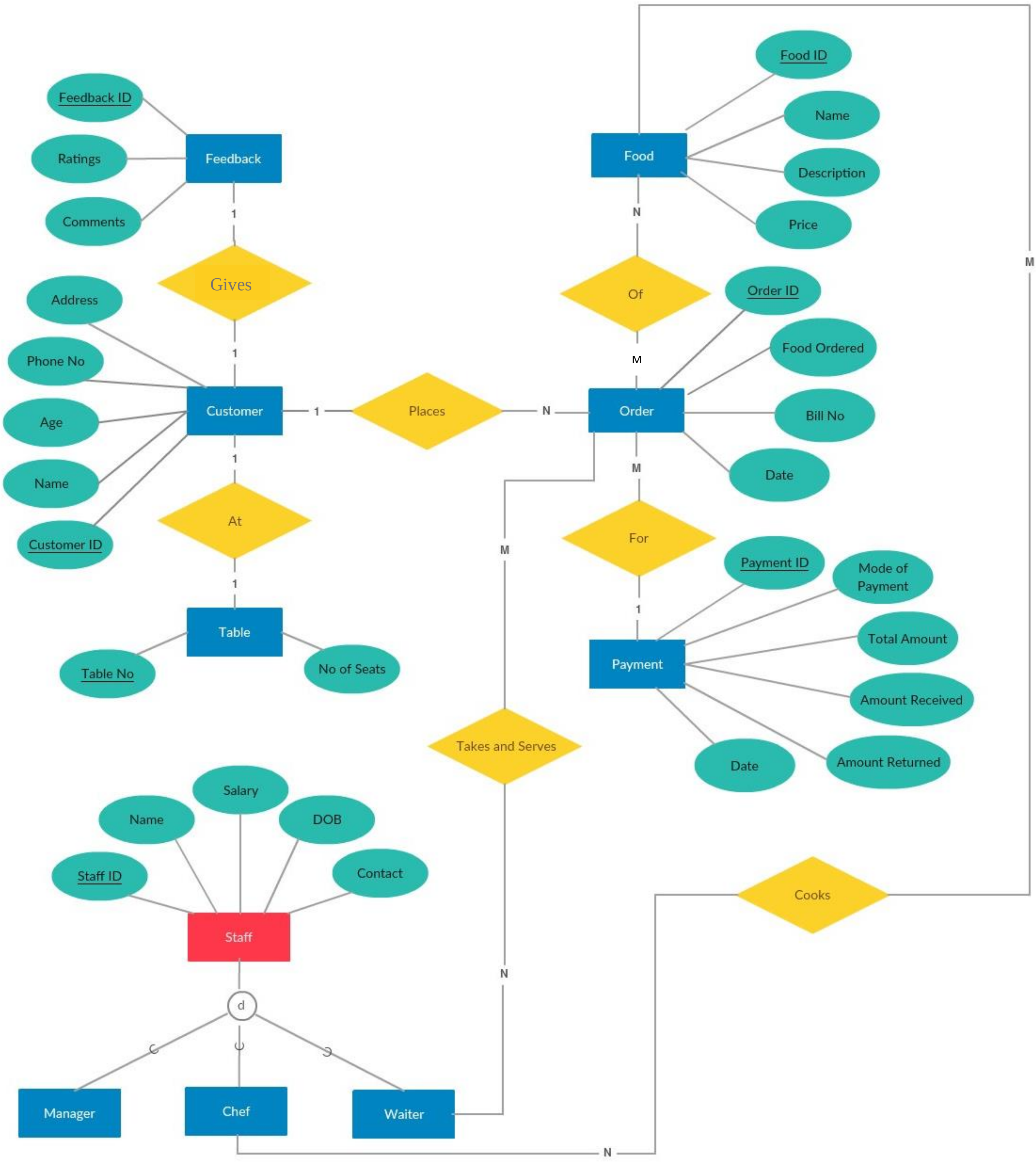
Modification of data

1. A customer has changed his address. So, his address needs to be updated for delivery of his order by order_id.
2. The amount of payment needs to be updated for an order
3. A customer decides to change his feedback for services of the restaurant
4. A customer decides to change his mobile number at time of delivery of an order

Data Retrieval

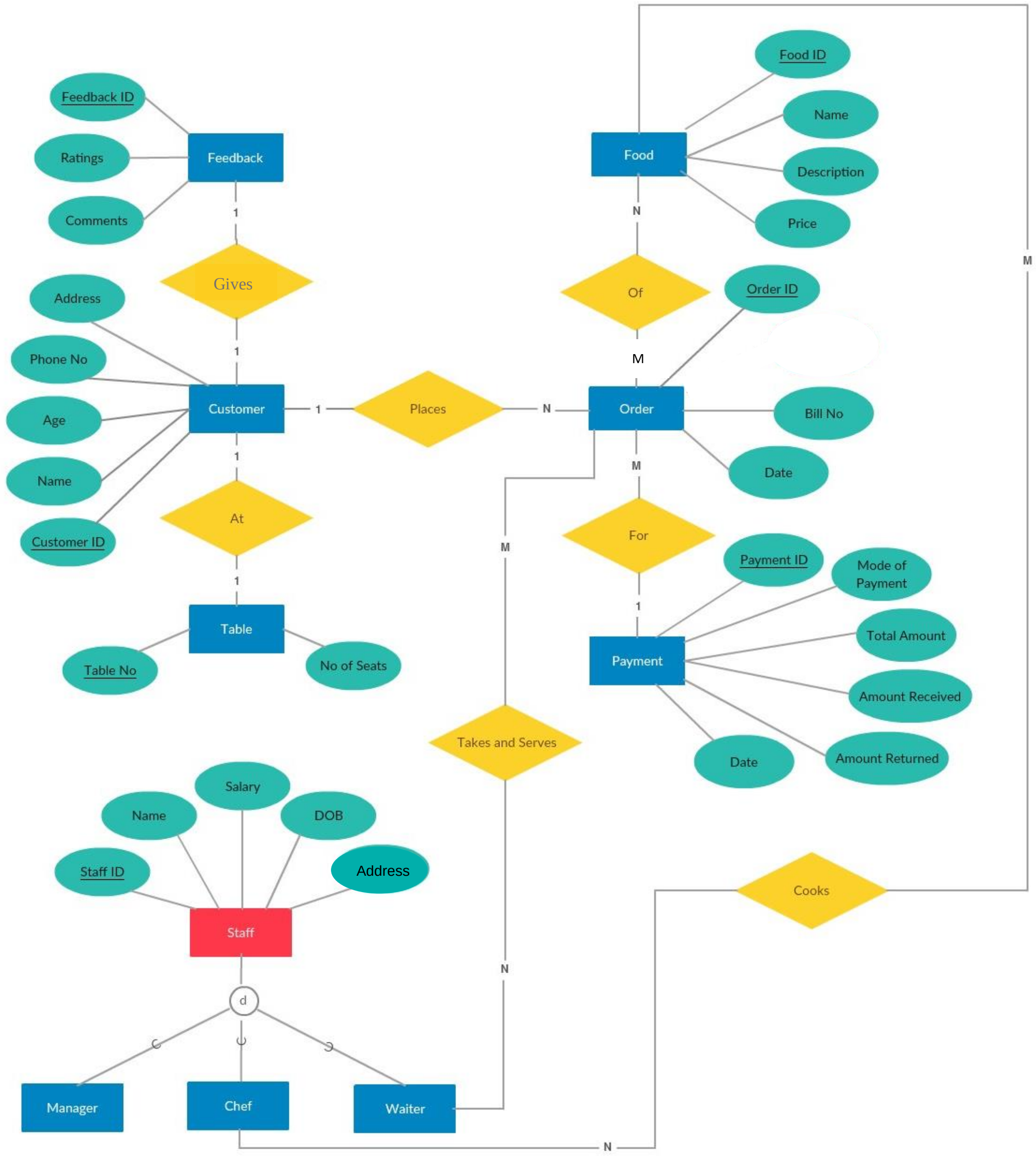
1. Display the name of the customer who gave order on the entered date.
2. Display the list of all the bill no for a particular food
3. Display the rating given by a particular customer
4. Display the name of the customer who did gave particular comment and rating
5. Display the table_no where a particular order was served

EER MODEL



Feedback ID
Ratings
Comments
Feedback
1
Gives
1
Address
Phone No
Age
Name
Customer ID
Customer
1
Places
N
Order
1
At
1
Table
Table No
No of Seats
Food
Food ID
Name
Description
Price
N
Of
M
Order ID
Food Ordered
Bill No
Date
M
For
1
Payment
Payment ID
Mode of Payment
Total Amount
Amount Received
Amount Returned
Date
M
Takes and Serves
N
M
Cooks
N
Salary
Name
DOB
Staff ID
Contact
Staff
d
Manager
Chef
Waiter

EER TO RELATIONAL DATABASE SCHEMA



1) Mapping Strong Entity types:

CUSTOMER(Customer_id, Name, Age, Phone_no, Address)

FEEDBACK(Feedback_id, Rating, Comments)

TABLE1(Table_no, No_of_seats)

PAYMENT(Payment_id, Mode_of_payment, Total_amount, Amount_recieved, Amount_returned, Date_of_payment)

ORDER1(Order_id,Bill_no,Date_of_order)

FOOD(Food_id, Name, Description, Price)

MANAGER(Staff_id, Name, Salary, Date_of_Birth, Address)

WAITER(Staff_id, Name, Salary, Date_of_Birth, Address)

CHEF(Staff_id, Name, Salary, Date_of_Birth, Address)

2) Mapping Weak Entity types:

There are no weak entity types here. So, no need of this step.

3) Mapping One to One and One to Many Relationship types:

CUSTOMER(Customer_id, Name, Age, Phone_no, Address, Feedback_id, Table_no)

Feedback_id is a foreign key that refers to FEEDBACK(Feedback_id)

ORDER1(Order_id,Bill_no,Date_of_order,Customer_id, Payment_id)

Customer_id is a foreign key that refers to CUSTOMER(Customer_id)
Payment_id is a foreign key that refers to PAYMENT(Payment_id)

4) Mapping Many to Many Relationship types:

ORDER_OF_FOOD(Order_id, Food_id)

Order_id is a foreign key that refers to ORDER(Order_id)
Food_id is a foreign key that refers to FOOD(Food_id)

TAKES_AND_SERVES(Staff_id, Order_id)

Staff_id is a foreign key that refers to WAITER(Staff_id)
Order_id is a foreign key that refers to ORDER(Order_id)

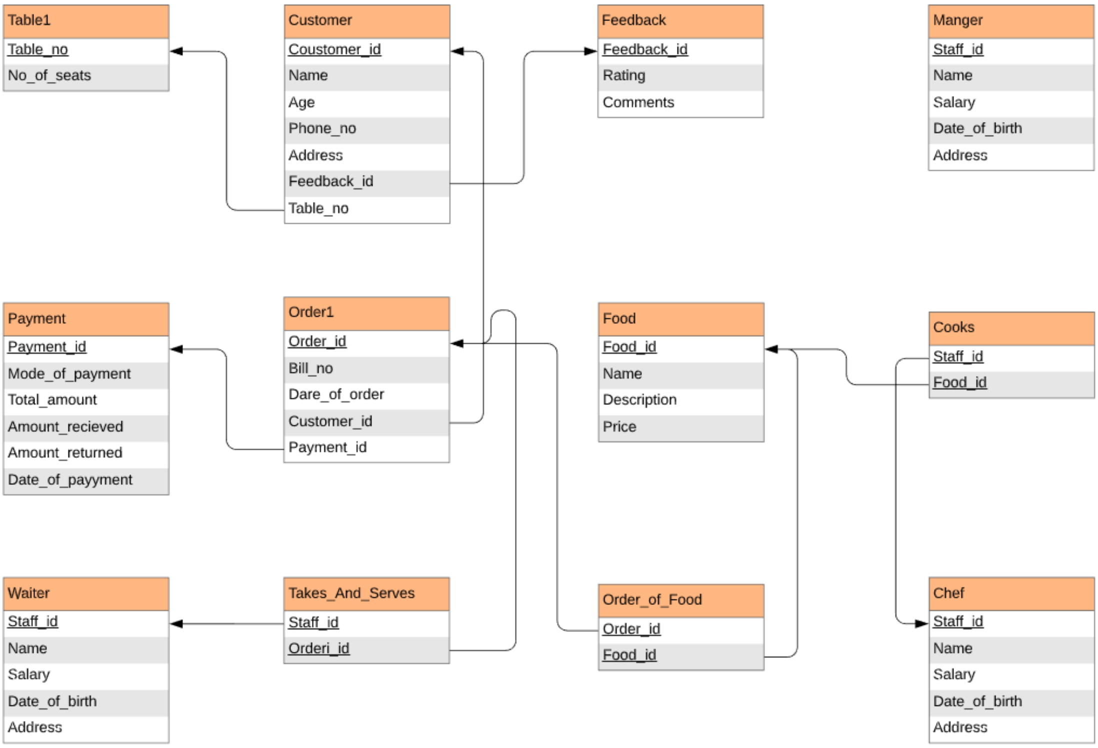
COOKS(Staff_id, Food_id)

Staff_id is a foreign key that refers to CHEF(Staff_id)
Food_id is a foreign key that refers to FOOD(Food_id)

5) Mapping Multi-valued Attributes:

There are no weak multi-valued attributes here. So, no need of this step.

RELATIONAL SCHEMA



DATA CONSTRAINTS

1. Customer

a. Customer_id:

PRIMARY KEY

b. Name:

NOT NULL

2. Order

a. Order_id :

PRIMARY KEY

b. BillNo :

UNIQUE

3. Payment

a. Payment_id

PRIMARY KEY

4. Waiter

a. Staff_id :

PRIMARY KEY

b. Name :

NOT NULL

c. Address:

NOT NULL

d. Check Salary>5000

5. Chef

a. Staff_id :

PRIMARY KEY

b. Name :

NOT NULL

c. Address:

NOT NULL

d. check Salary>5000

6. Manager

a. Manager_id :

PRIMARY KEY

b. Name :

NOT NULL

c. Address:

NOT NULL

d. Check Salary>20000

7. Food

a. Food_id :

PRIMARY KEY

8. Tables

a. TableNo

PRIMARY KEY

9. Feedback

a. Feedback_id

PRIMARY KEY

10. Takes_and_Serves

a. Staff_id and Order_id

PRIMARY KEY

11. Order_of_Food

a. Order_id and Food_id

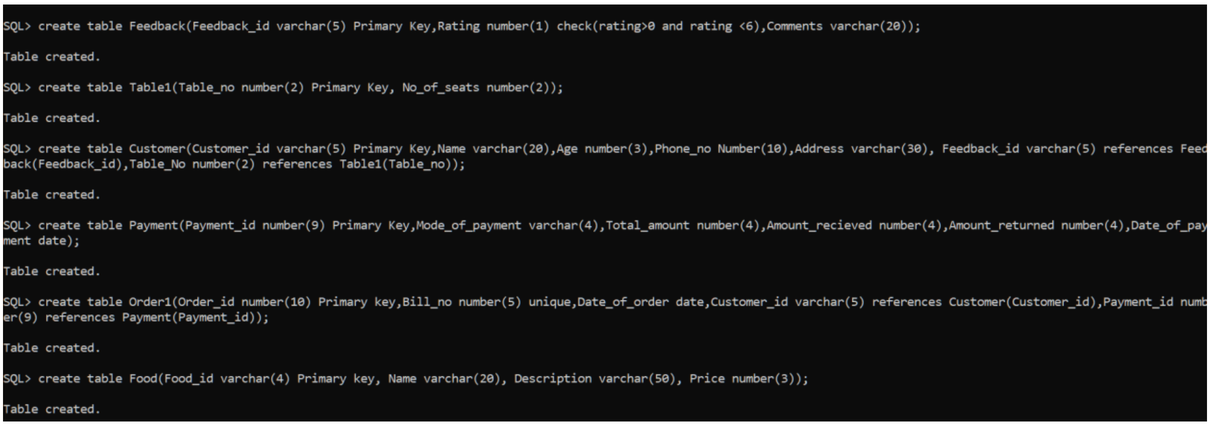
PRIMARY KEY

12. Cooks

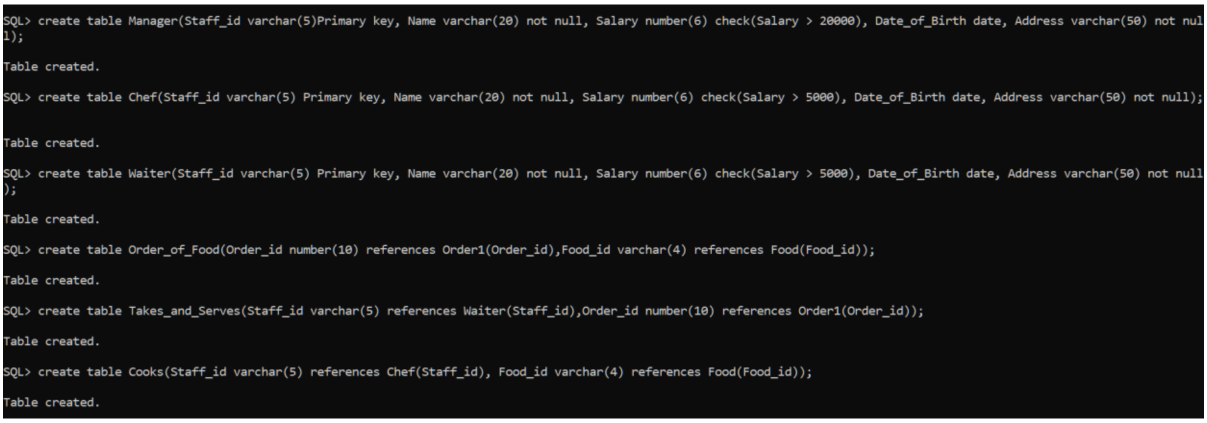
a. Staff_id and Food_id

PRIMARY KEY

TABLE CREATIONS WITH CONSTAINTS AND KEYS



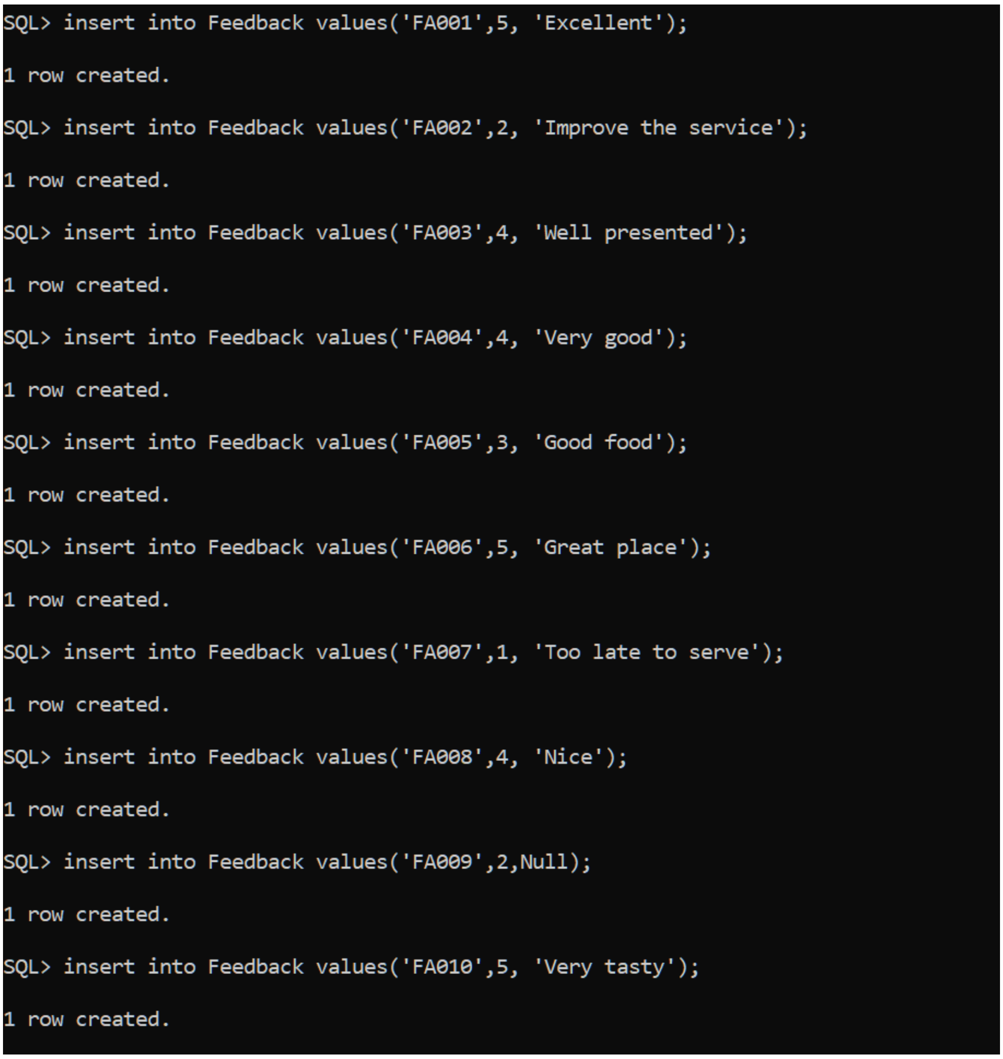
```
SQL> create table Feedback(Feedback_id varchar(5) Primary Key,Rating number(1) check(rating>0 and rating <6),Comments varchar(20));

Table created.

SQL> create table Table1(Table_no number(2) Primary Key, No_of_seats number(2));

Table created.

SQL> create table Customer(Customer_id varchar(5) Primary Key,Name varchar(20),Age number(3),Phone_no Number(10),Address varchar(30), Feedback_id varchar(5) references Feed
back(Feedback_id),Table_No number(2) references Table1(Table_no));

Table created.

SQL> create table Payment(Payment_id number(9) Primary Key,Mode_of_payment varchar(4),Total_amount number(4),Amount_recieved number(4),Amount_returned number(4),Date_of_pay
ment date);

Table created.

SQL> create table Order1(Order_id number(10) Primary key,Bill_no number(5) unique,Date_of_order date,Customer_id varchar(5) references Customer(Customer_id),Payment_id numb
er(9) references Payment(Payment_id));

Table created.

SQL> create table Food(Food_id varchar(4) Primary key, Name varchar(20), Description varchar(50), Price number(3));

Table created.
```



```
SQL> create table Manager(Staff_id varchar(5)Primary key, Name varchar(20) not null, Salary number(6) check(Salary > 20000), Date_of_Birth date, Address varchar(50) not nul
l);

Table created.

SQL> create table Chef(Staff_id varchar(5) Primary key, Name varchar(20) not null, Salary number(6) check(Salary > 5000), Date_of_Birth date, Address varchar(50) not null);


Table created.

SQL> create table Waiter(Staff_id varchar(5) Primary key, Name varchar(20) not null, Salary number(6) check(Salary > 5000), Date_of_Birth date, Address varchar(50) not null
);

Table created.

SQL> create table Order_of_Food(Order_id number(10) references Order1(Order_id),Food_id varchar(4) references Food(Food_id));

Table created.

SQL> create table Takes_and_Serves(Staff_id varchar(5) references Waiter(Staff_id),Order_id number(10) references Order1(Order_id));

Table created.

SQL> create table Cooks(Staff_id varchar(5) references Chef(Staff_id), Food_id varchar(4) references Food(Food_id));

Table created.
```

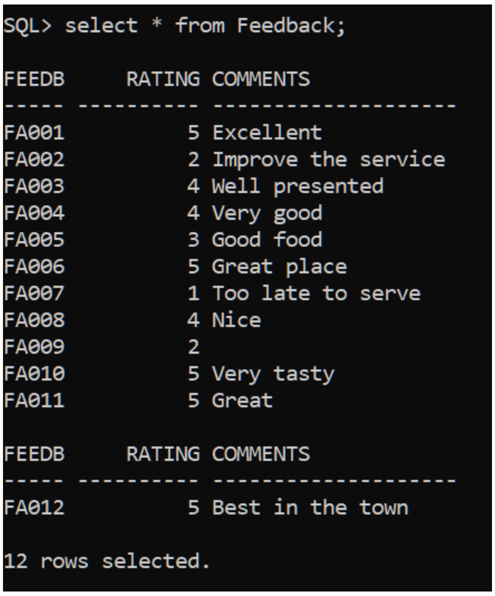
DATA INSERTIONS

Feedback Table



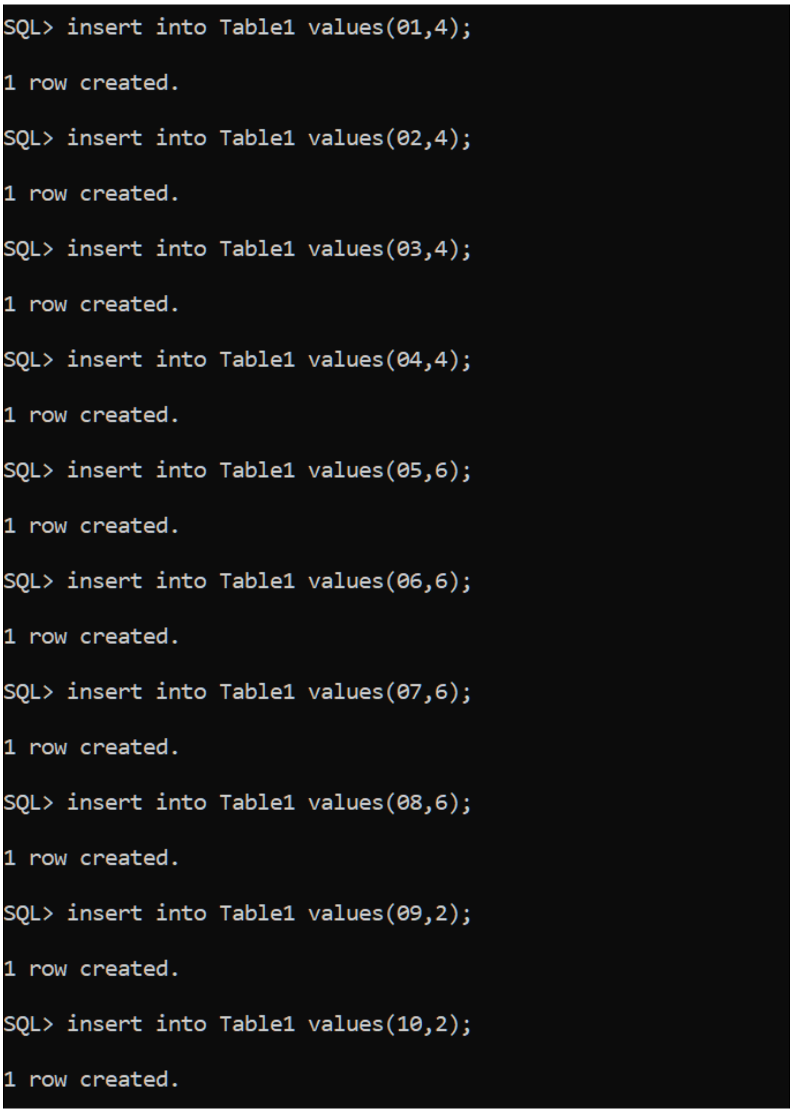
```
SQL> insert into Feedback values('FA001',5, 'Excellent');

1 row created.

SQL> insert into Feedback values('FA002',2, 'Improve the service');

1 row created.

SQL> insert into Feedback values('FA003',4, 'Well presented');

1 row created.

SQL> insert into Feedback values('FA004',4, 'Very good');

1 row created.

SQL> insert into Feedback values('FA005',3, 'Good food');

1 row created.

SQL> insert into Feedback values('FA006',5, 'Great place');

1 row created.

SQL> insert into Feedback values('FA007',1, 'Too late to serve');

1 row created.

SQL> insert into Feedback values('FA008',4, 'Nice');

1 row created.

SQL> insert into Feedback values('FA009',2,Null);

1 row created.

SQL> insert into Feedback values('FA010',5, 'Very tasty');

1 row created.
```

Final Table



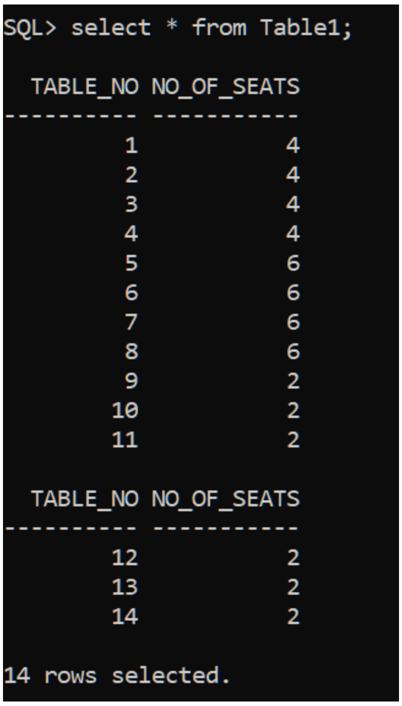
```
SQL> select * from Feedback;

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA001          5 Excellent
FA002          2 Improve the service
FA003          4 Well presented
FA004          4 Very good
FA005          3 Good food
FA006          5 Great place
FA007          1 Too late to serve
FA008          4 Nice
FA009          2
FA010          5 Very tasty
FA011          5 Great

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA012          5 Best in the town

12 rows selected.
```

Table1 Table



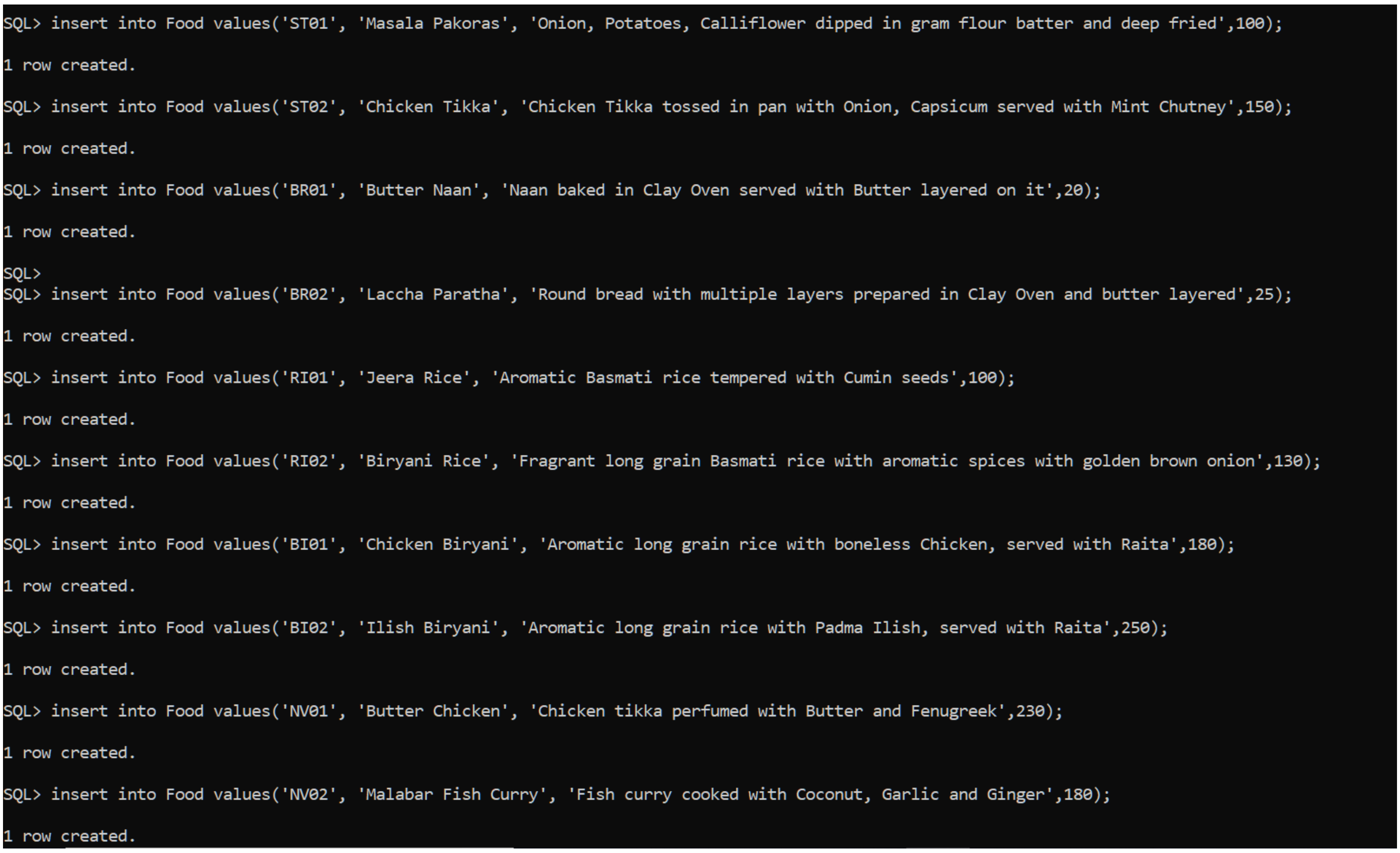
```
SQL> insert into Table1 values(01,4);

1 row created.

SQL> insert into Table1 values(02,4);

1 row created.

SQL> insert into Table1 values(03,4);

1 row created.

SQL> insert into Table1 values(04,4);

1 row created.

SQL> insert into Table1 values(05,6);

1 row created.

SQL> insert into Table1 values(06,6);

1 row created.

SQL> insert into Table1 values(07,6);

1 row created.

SQL> insert into Table1 values(08,6);

1 row created.

SQL> insert into Table1 values(09,2);

1 row created.

SQL> insert into Table1 values(10,2);

1 row created.
```

Final Table



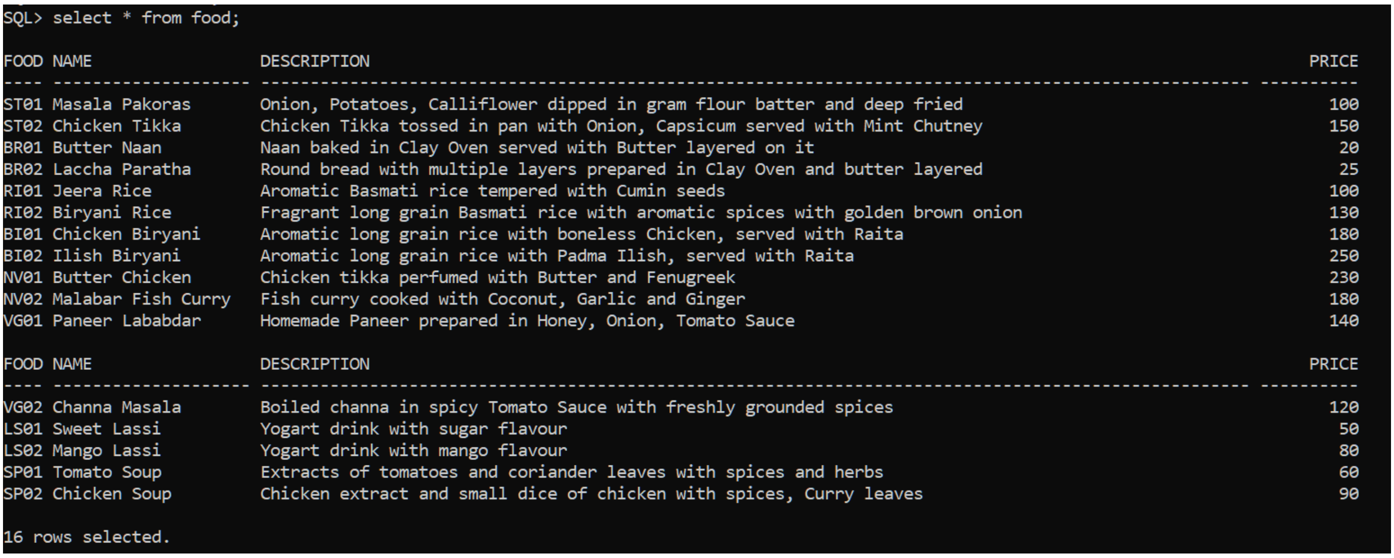
```
SQL> select * from Table1;

  TABLE_NO NO_OF_SEATS
---------- -----------
         1           4
         2           4
         3           4
         4           4
         5           6
         6           6
         7           6
         8           6
         9           2
        10           2
        11           2

  TABLE_NO NO_OF_SEATS
---------- -----------
        12           2
        13           2
        14           2

14 rows selected.
```

Food Table



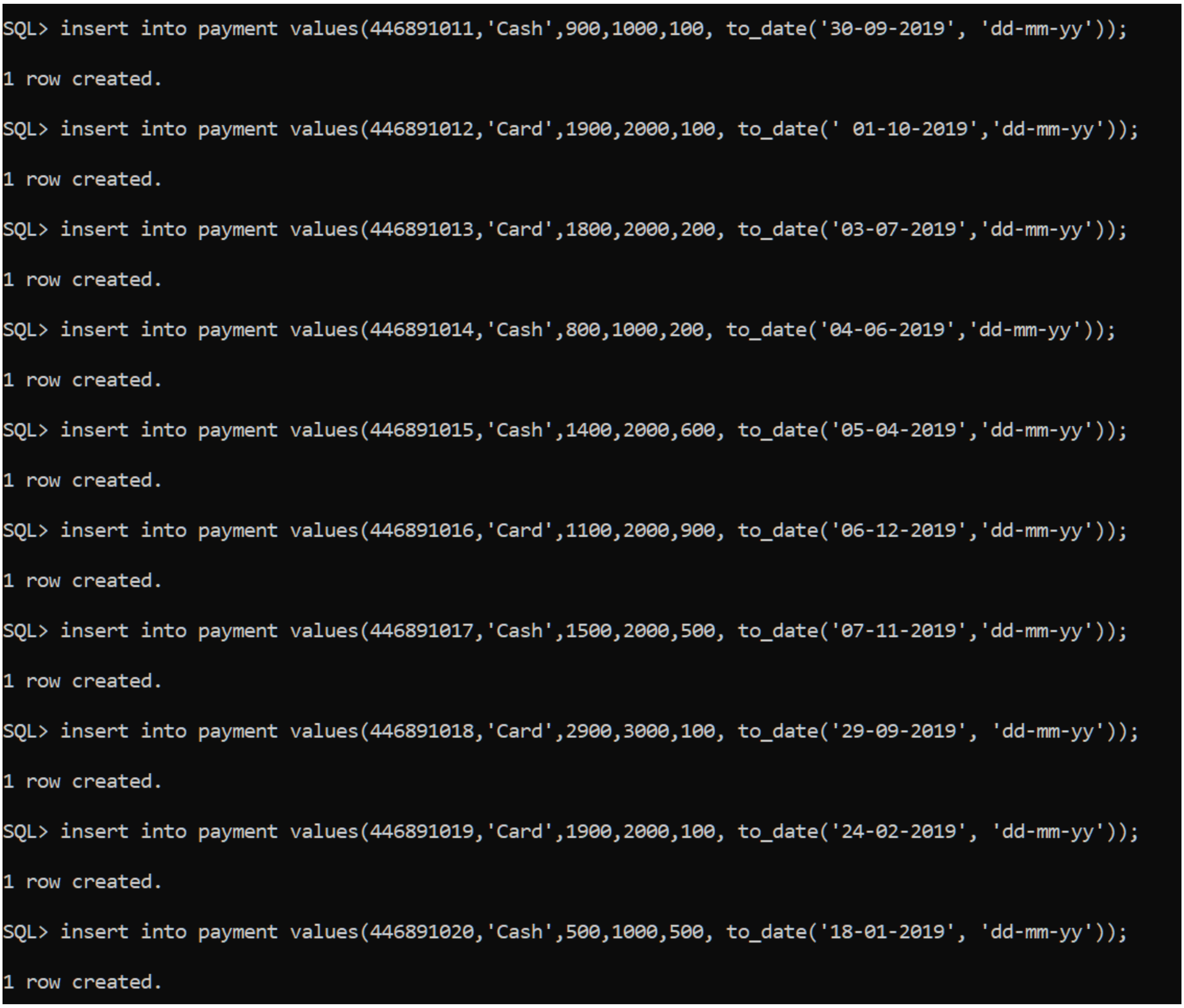
```
SQL> insert into Food values('ST01', 'Masala Pakoras', 'Onion, Potatoes, Calliflower dipped in gram flour batter and deep fried',100);

1 row created.

SQL> insert into Food values('ST02', 'Chicken Tikka', 'Chicken Tikka tossed in pan with Onion, Capsicum served with Mint Chutney',150);

1 row created.

SQL> insert into Food values('BR01', 'Butter Naan', 'Naan baked in Clay Oven served with Butter layered on it',20);

1 row created.

SQL>
SQL> insert into Food values('BR02', 'Laccha Paratha', 'Round bread with multiple layers prepared in Clay Oven and butter layered',25);

1 row created.

SQL> insert into Food values('RI01', 'Jeera Rice', 'Aromatic Basmati rice tempered with Cumin seeds',100);

1 row created.

SQL> insert into Food values('RI02', 'Biryani Rice', 'Fragrant long grain Basmati rice with aromatic spices with golden brown onion',130);

1 row created.

SQL> insert into Food values('BI01', 'Chicken Biryani', 'Aromatic long grain rice with boneless Chicken, served with Raita',180);

1 row created.

SQL> insert into Food values('BI02', 'Ilish Biryani', 'Aromatic long grain rice with Padma Ilish, served with Raita',250);

1 row created.

SQL> insert into Food values('NV01', 'Butter Chicken', 'Chicken tikka perfumed with Butter and Fenugreek',230);

1 row created.

SQL> insert into Food values('NV02', 'Malabar Fish Curry', 'Fish curry cooked with Coconut, Garlic and Ginger',180);

1 row created.
```

Final Table



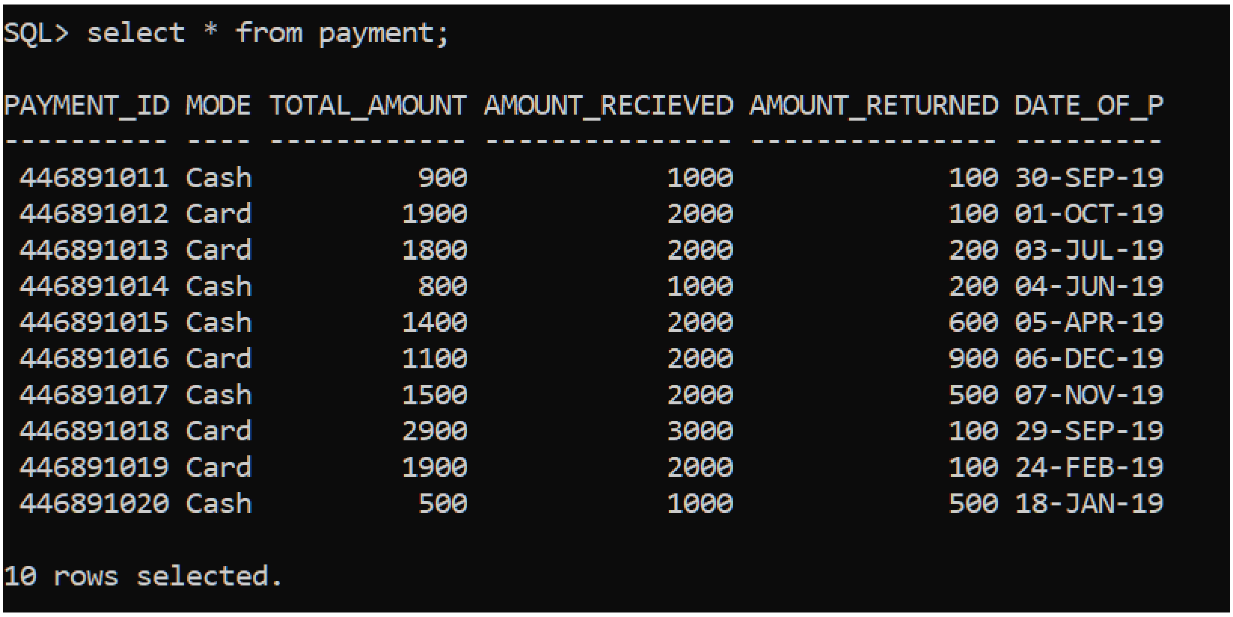
```
SQL> select * from food;

FOOD NAME                 DESCRIPTION                                                                                               PRICE
---- -------------------- ---------------------------------------------------------------------------------------------------- ----------
ST01 Masala Pakoras       Onion, Potatoes, Calliflower dipped in gram flour batter and deep fried                                     100
ST02 Chicken Tikka        Chicken Tikka tossed in pan with Onion, Capsicum served with Mint Chutney                                   150
BR01 Butter Naan          Naan baked in Clay Oven served with Butter layered on it                                                     20
BR02 Laccha Paratha       Round bread with multiple layers prepared in Clay Oven and butter layered                                    25
RI01 Jeera Rice           Aromatic Basmati rice tempered with Cumin seeds                                                             100
RI02 Biryani Rice         Fragrant long grain Basmati rice with aromatic spices with golden brown onion                               130
BI01 Chicken Biryani      Aromatic long grain rice with boneless Chicken, served with Raita                                           180
BI02 Ilish Biryani        Aromatic long grain rice with Padma Ilish, served with Raita                                                250
NV01 Butter Chicken       Chicken tikka perfumed with Butter and Fenugreek                                                            230
NV02 Malabar Fish Curry   Fish curry cooked with Coconut, Garlic and Ginger                                                           180
VG01 Paneer Lababdar      Homemade Paneer prepared in Honey, Onion, Tomato Sauce                                                      140

FOOD NAME                 DESCRIPTION                                                                                               PRICE
---- -------------------- ---------------------------------------------------------------------------------------------------- ----------
VG02 Channa Masala        Boiled channa in spicy Tomato Sauce with freshly grounded spices                                            120
LS01 Sweet Lassi          Yogart drink with sugar flavour                                                                              50
LS02 Mango Lassi          Yogart drink with mango flavour                                                                              80
SP01 Tomato Soup          Extracts of tomatoes and coriander leaves with spices and herbs                                              60
SP02 Chicken Soup         Chicken extract and small dice of chicken with spices, Curry leaves                                          90

16 rows selected.
```

Payment Table



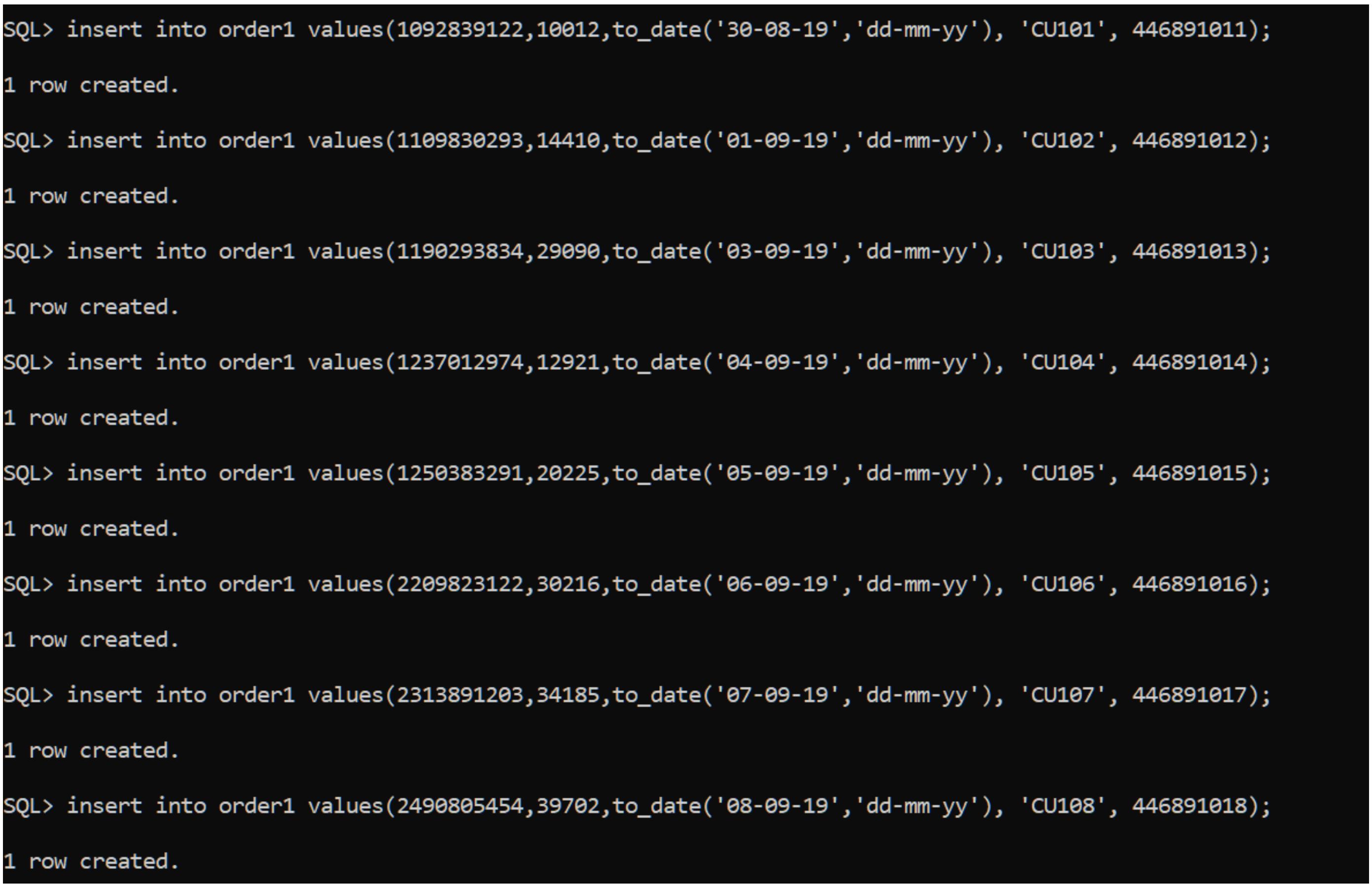
```
SQL> insert into payment values(446891011,'Cash',900,1000,100, to_date('30-09-2019', 'dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891012,'Card',1900,2000,100, to_date(' 01-10-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891013,'Card',1800,2000,200, to_date('03-07-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891014,'Cash',800,1000,200, to_date('04-06-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891015,'Cash',1400,2000,600, to_date('05-04-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891016,'Card',1100,2000,900, to_date('06-12-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891017,'Cash',1500,2000,500, to_date('07-11-2019','dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891018,'Card',2900,3000,100, to_date('29-09-2019', 'dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891019,'Card',1900,2000,100, to_date('24-02-2019', 'dd-mm-yy'));

1 row created.

SQL> insert into payment values(446891020,'Cash',500,1000,500, to_date('18-01-2019', 'dd-mm-yy'));

1 row created.
```

Final Table



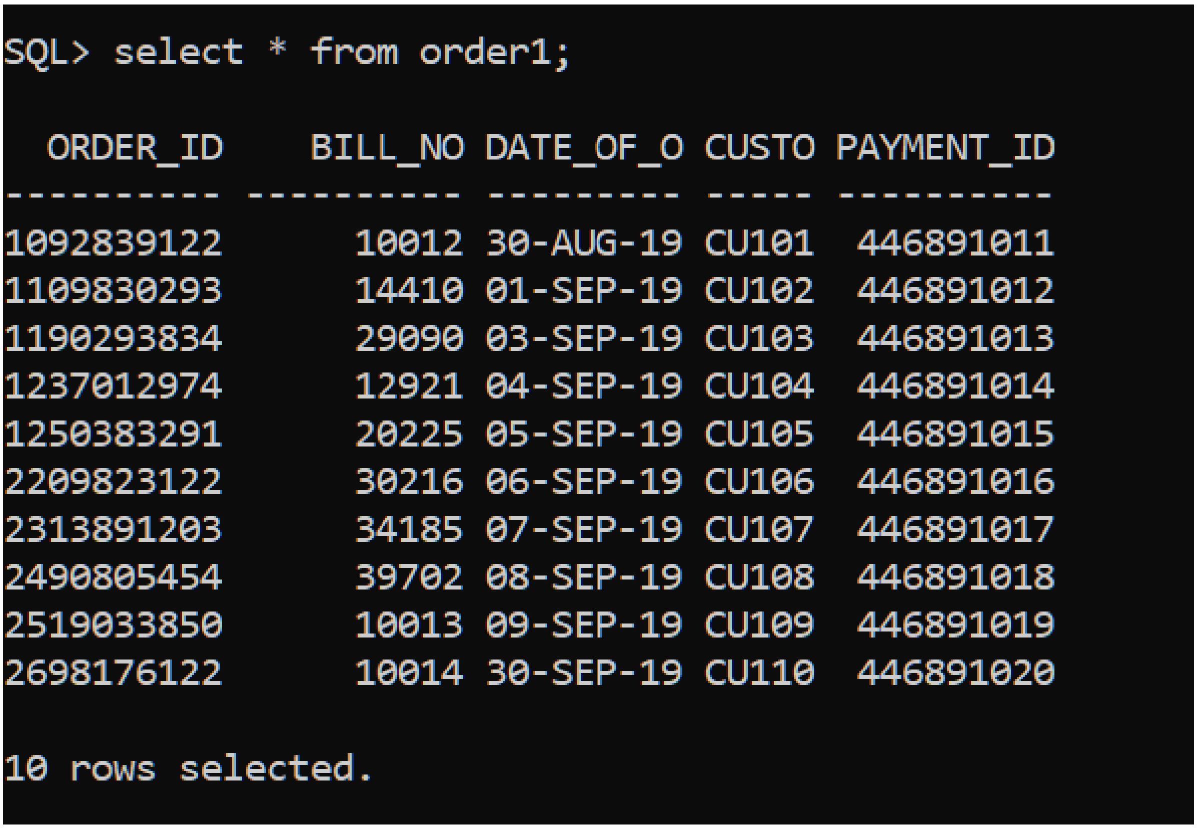
```
SQL> select * from payment;

PAYMENT_ID MODE TOTAL_AMOUNT AMOUNT_RECIEVED AMOUNT_RETURNED DATE_OF_P
---------- ---- ------------ --------------- --------------- ---------
 446891011 Cash          900            1000             100 30-SEP-19
 446891012 Card         1900            2000             100 01-OCT-19
 446891013 Card         1800            2000             200 03-JUL-19
 446891014 Cash          800            1000             200 04-JUN-19
 446891015 Cash         1400            2000             600 05-APR-19
 446891016 Card         1100            2000             900 06-DEC-19
 446891017 Cash         1500            2000             500 07-NOV-19
 446891018 Card         2900            3000             100 29-SEP-19
 446891019 Card         1900            2000             100 24-FEB-19
 446891020 Cash          500            1000             500 18-JAN-19

10 rows selected.
```

Order1 Table



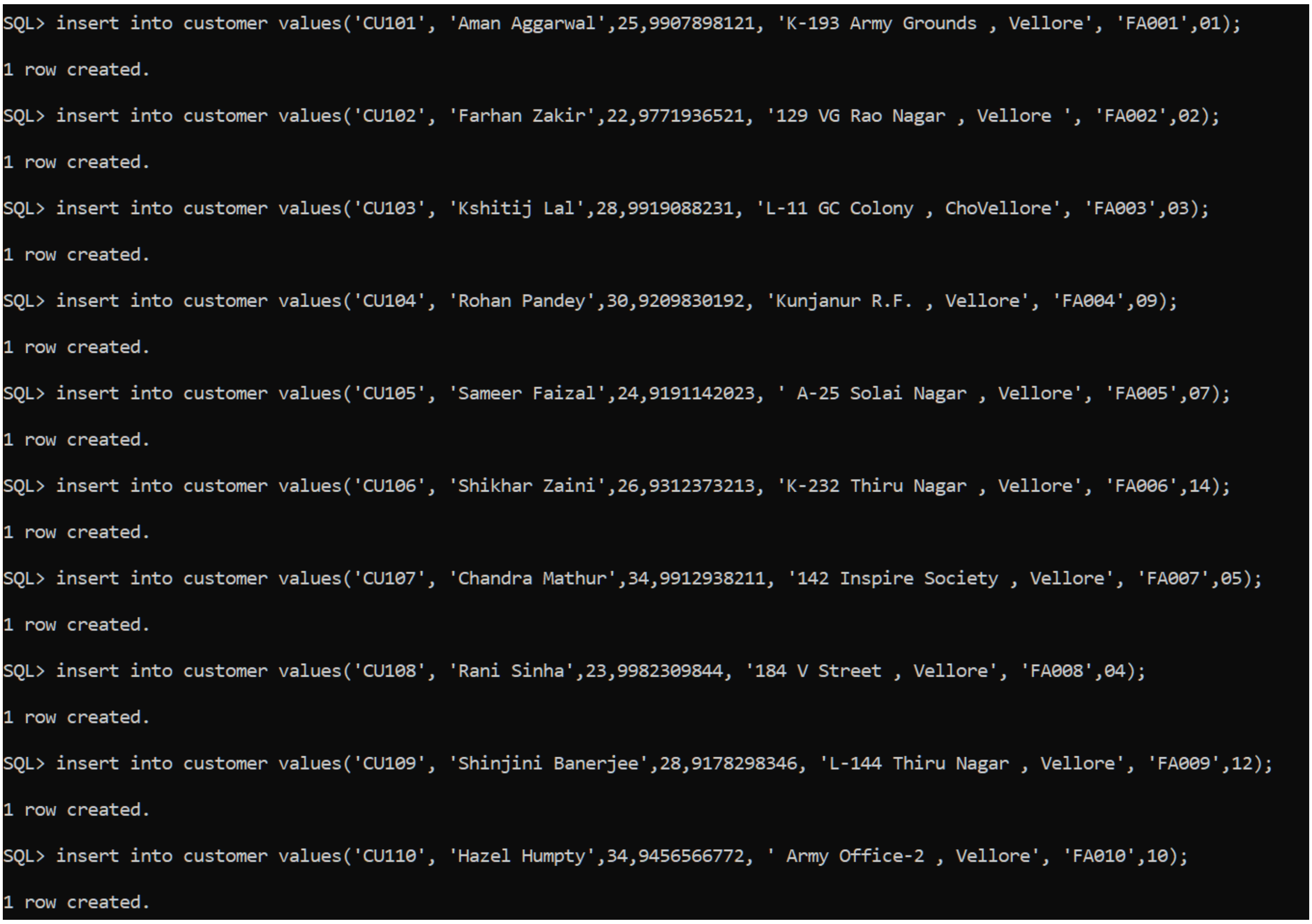
```
SQL> insert into order1 values(1092839122,10012,to_date('30-08-19','dd-mm-yy'), 'CU101', 446891011);

1 row created.

SQL> insert into order1 values(1109830293,14410,to_date('01-09-19','dd-mm-yy'), 'CU102', 446891012);

1 row created.

SQL> insert into order1 values(1190293834,29090,to_date('03-09-19','dd-mm-yy'), 'CU103', 446891013);

1 row created.

SQL> insert into order1 values(1237012974,12921,to_date('04-09-19','dd-mm-yy'), 'CU104', 446891014);

1 row created.

SQL> insert into order1 values(1250383291,20225,to_date('05-09-19','dd-mm-yy'), 'CU105', 446891015);

1 row created.

SQL> insert into order1 values(2209823122,30216,to_date('06-09-19','dd-mm-yy'), 'CU106', 446891016);

1 row created.

SQL> insert into order1 values(2313891203,34185,to_date('07-09-19','dd-mm-yy'), 'CU107', 446891017);

1 row created.

SQL> insert into order1 values(2490805454,39702,to_date('08-09-19','dd-mm-yy'), 'CU108', 446891018);

1 row created.
```

Final Table



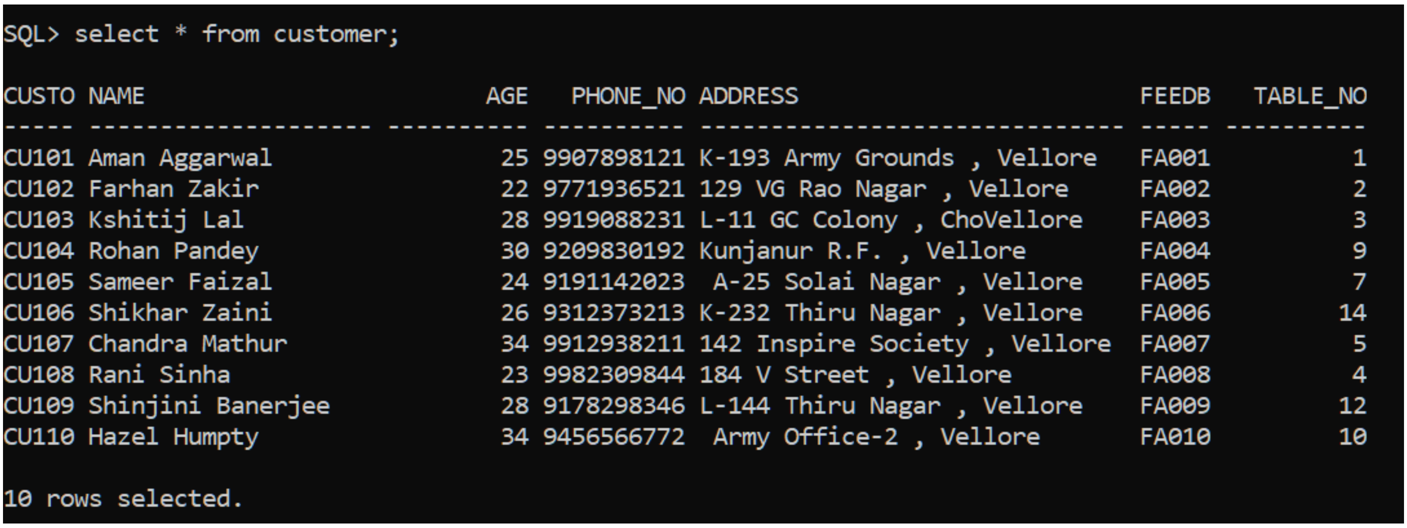
```
SQL> select * from order1;

  ORDER_ID    BILL_NO DATE_OF_O CUSTO PAYMENT_ID
---------- ---------- --------- ----- ----------
1092839122      10012 30-AUG-19 CU101  446891011
1109830293      14410 01-SEP-19 CU102  446891012
1190293834      29090 03-SEP-19 CU103  446891013
1237012974      12921 04-SEP-19 CU104  446891014
1250383291      20225 05-SEP-19 CU105  446891015
2209823122      30216 06-SEP-19 CU106  446891016
2313891203      34185 07-SEP-19 CU107  446891017
2490805454      39702 08-SEP-19 CU108  446891018
2519033850      10013 09-SEP-19 CU109  446891019
2698176122      10014 30-SEP-19 CU110  446891020

10 rows selected.
```

Customer Table



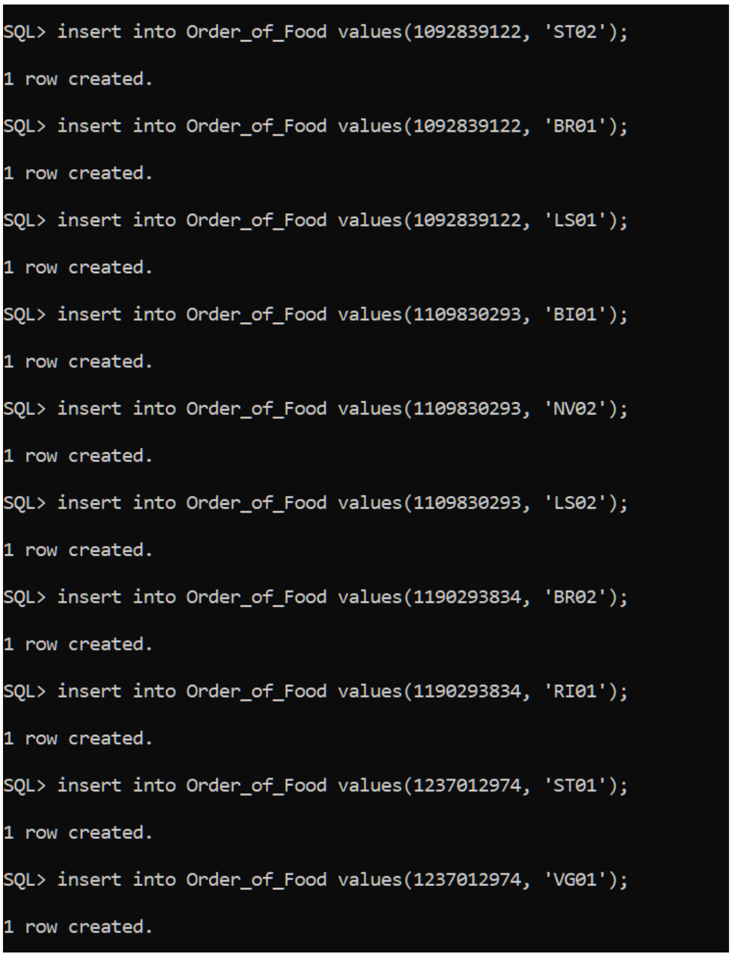
```
SQL> insert into customer values('CU101', 'Aman Aggarwal',25,9907898121, 'K-193 Army Grounds , Vellore', 'FA001',01);

1 row created.

SQL> insert into customer values('CU102', 'Farhan Zakir',22,9771936521, '129 VG Rao Nagar , Vellore ', 'FA002',02);

1 row created.

SQL> insert into customer values('CU103', 'Kshitij Lal',28,9919088231, 'L-11 GC Colony , ChoVellore', 'FA003',03);

1 row created.

SQL> insert into customer values('CU104', 'Rohan Pandey',30,9209830192, 'Kunjanur R.F. , Vellore', 'FA004',09);

1 row created.

SQL> insert into customer values('CU105', 'Sameer Faizal',24,9191142023, ' A-25 Solai Nagar , Vellore', 'FA005',07);

1 row created.

SQL> insert into customer values('CU106', 'Shikhar Zaini',26,9312373213, 'K-232 Thiru Nagar , Vellore', 'FA006',14);

1 row created.

SQL> insert into customer values('CU107', 'Chandra Mathur',34,9912938211, '142 Inspire Society , Vellore', 'FA007',05);

1 row created.

SQL> insert into customer values('CU108', 'Rani Sinha',23,9982309844, '184 V Street , Vellore', 'FA008',04);

1 row created.

SQL> insert into customer values('CU109', 'Shinjini Banerjee',28,9178298346, 'L-144 Thiru Nagar , Vellore', 'FA009',12);

1 row created.

SQL> insert into customer values('CU110', 'Hazel Humpty',34,9456566772, ' Army Office-2 , Vellore', 'FA010',10);

1 row created.
```

Final Table



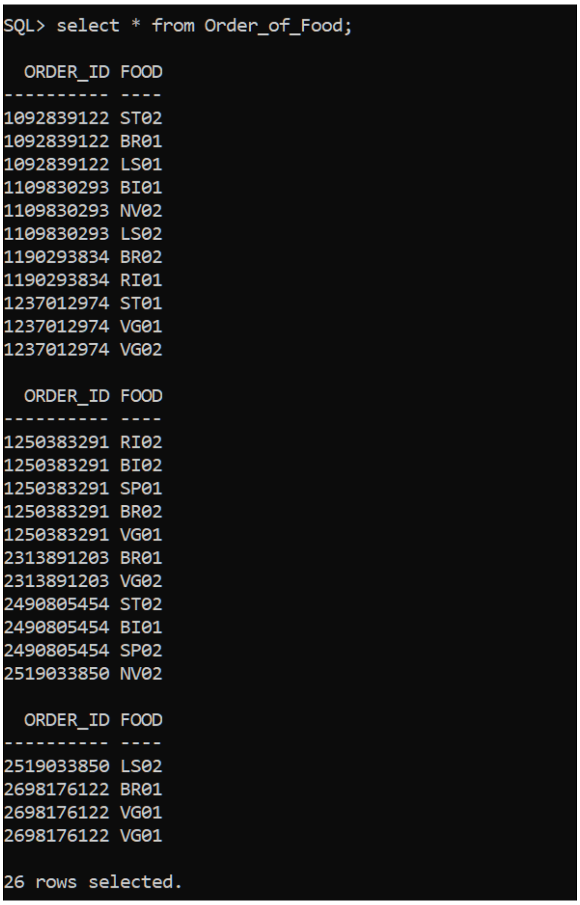
```
SQL> select * from customer;

CUSTO NAME                        AGE   PHONE_NO ADDRESS                        FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ------------------------------ ----- ----------
CU101 Aman Aggarwal                25 9907898121 K-193 Army Grounds , Vellore   FA001          1
CU102 Farhan Zakir                 22 9771936521 129 VG Rao Nagar , Vellore     FA002          2
CU103 Kshitij Lal                  28 9919088231 L-11 GC Colony , ChoVellore    FA003          3
CU104 Rohan Pandey                 30 9209830192 Kunjanur R.F. , Vellore        FA004          9
CU105 Sameer Faizal                24 9191142023  A-25 Solai Nagar , Vellore    FA005          7
CU106 Shikhar Zaini                26 9312373213 K-232 Thiru Nagar , Vellore    FA006         14
CU107 Chandra Mathur               34 9912938211 142 Inspire Society , Vellore  FA007          5
CU108 Rani Sinha                   23 9982309844 184 V Street , Vellore         FA008          4
CU109 Shinjini Banerjee            28 9178298346 L-144 Thiru Nagar , Vellore    FA009         12
CU110 Hazel Humpty                 34 9456566772  Army Office-2 , Vellore       FA010         10

10 rows selected.
```

Order_of_Food Table



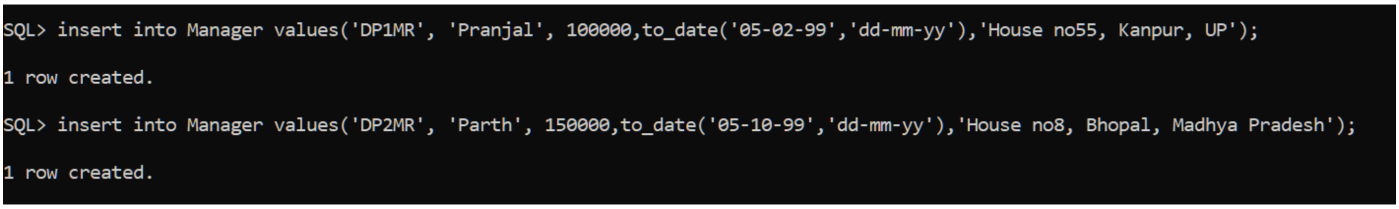
```
SQL> insert into Order_of_Food values(1092839122, 'ST02');

1 row created.

SQL> insert into Order_of_Food values(1092839122, 'BR01');

1 row created.

SQL> insert into Order_of_Food values(1092839122, 'LS01');

1 row created.

SQL> insert into Order_of_Food values(1109830293, 'BI01');

1 row created.

SQL> insert into Order_of_Food values(1109830293, 'NV02');

1 row created.

SQL> insert into Order_of_Food values(1109830293, 'LS02');

1 row created.

SQL> insert into Order_of_Food values(1190293834, 'BR02');

1 row created.

SQL> insert into Order_of_Food values(1190293834, 'RI01');

1 row created.

SQL> insert into Order_of_Food values(1237012974, 'ST01');

1 row created.

SQL> insert into Order_of_Food values(1237012974, 'VG01');

1 row created.
```

Final Table



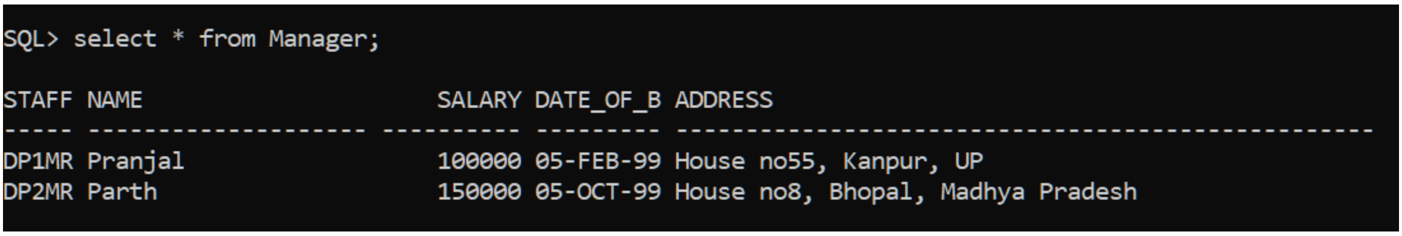
```
SQL> select * from Order_of_Food;

  ORDER_ID FOOD
---------- ----
1092839122 ST02
1092839122 BR01
1092839122 LS01
1109830293 BI01
1109830293 NV02
1109830293 LS02
1190293834 BR02
1190293834 RI01
1237012974 ST01
1237012974 VG01
1237012974 VG02

  ORDER_ID FOOD
---------- ----
1250383291 RI02
1250383291 BI02
1250383291 SP01
1250383291 BR02
1250383291 VG01
2313891203 BR01
2313891203 VG02
2490805454 ST02
2490805454 BI01
2490805454 SP02
2519033850 NV02

  ORDER_ID FOOD
---------- ----
2519033850 LS02
2698176122 BR01
2698176122 VG01
2698176122 VG01

26 rows selected.
```

Manager Table



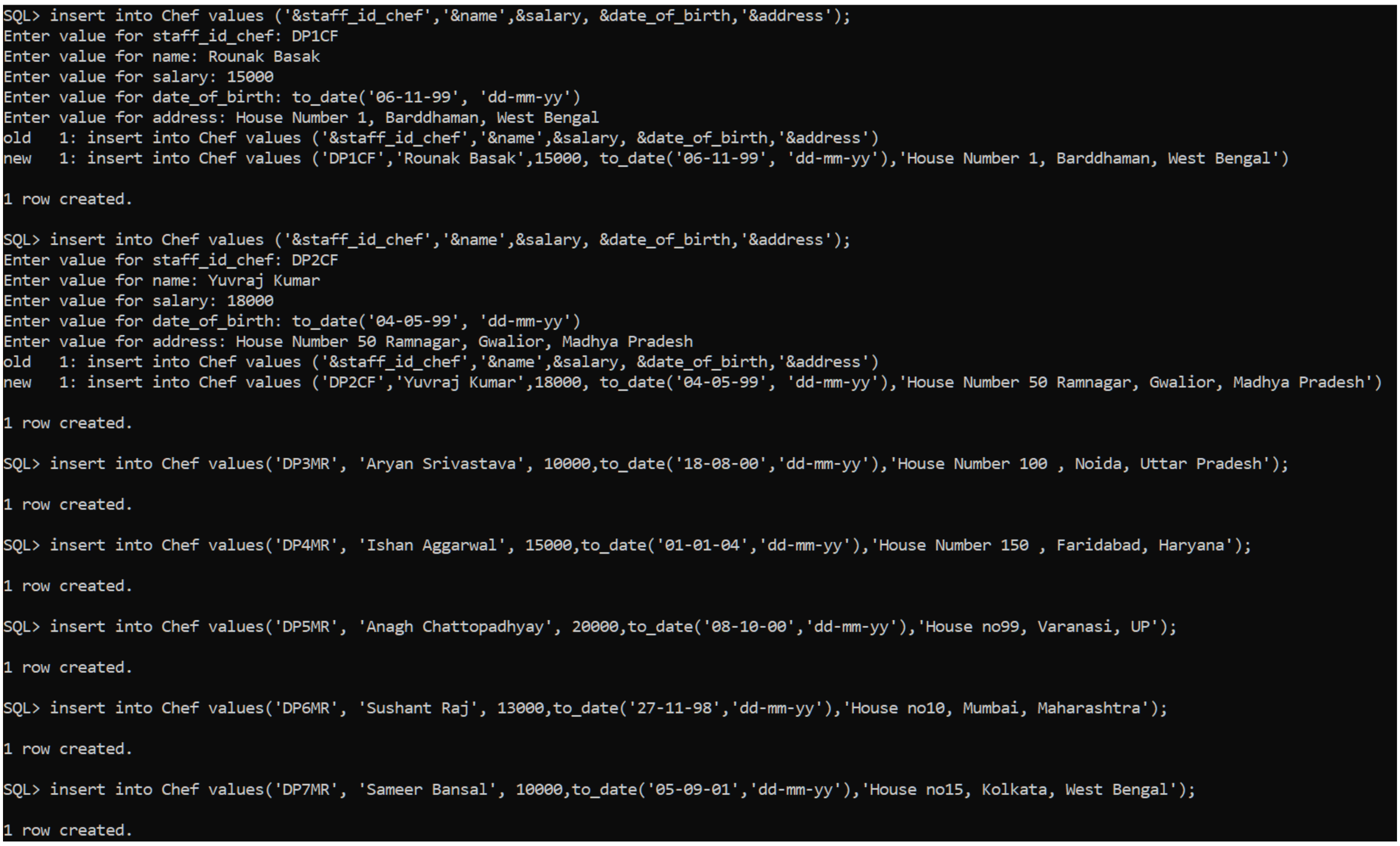
```
SQL> insert into Manager values('DP1MR', 'Pranjal', 100000,to_date('05-02-99','dd-mm-yy'),'House no55, Kanpur, UP');

1 row created.

SQL> insert into Manager values('DP2MR', 'Parth', 150000,to_date('05-10-99','dd-mm-yy'),'House no8, Bhopal, Madhya Pradesh');

1 row created.
```

Final Table



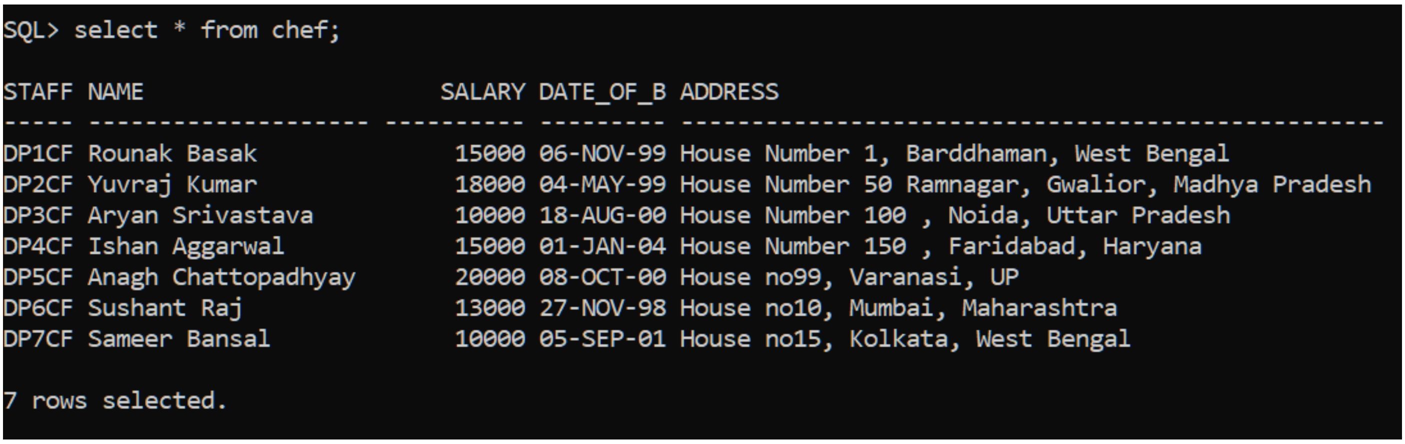
```
SQL> select * from Manager;

STAFF NAME                     SALARY DATE_OF_B ADDRESS
----- -------------------- ---------- --------- --------------------------------------------------
DP1MR Pranjal                  100000 05-FEB-99 House no55, Kanpur, UP
DP2MR Parth                    150000 05-OCT-99 House no8, Bhopal, Madhya Pradesh
```

Chef Table



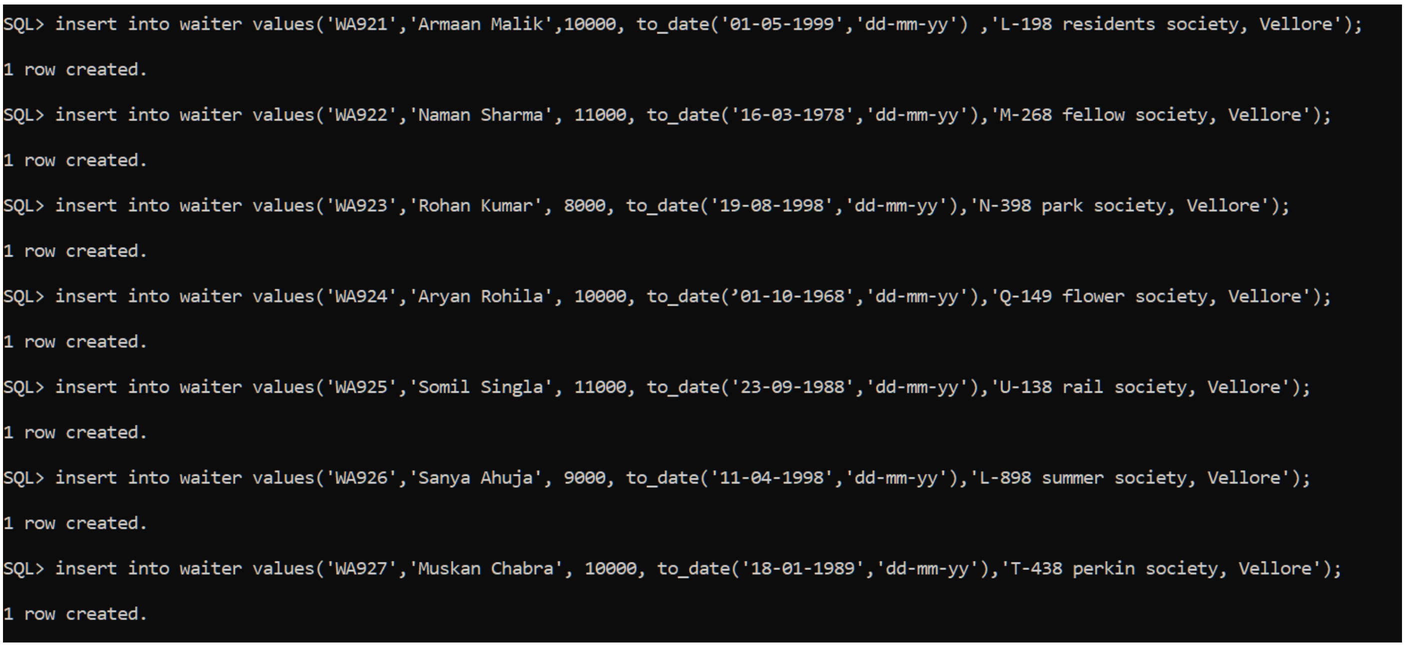
```
SQL> insert into Chef values ('&staff_id_chef','&name',&salary, &date_of_birth,'&address');
Enter value for staff_id_chef: DP1CF
Enter value for name: Rounak Basak
Enter value for salary: 15000
Enter value for date_of_birth: to_date('06-11-99', 'dd-mm-yy')
Enter value for address: House Number 1, Barddhaman, West Bengal
old   1: insert into Chef values ('&staff_id_chef','&name',&salary, &date_of_birth,'&address')
new   1: insert into Chef values ('DP1CF','Rounak Basak',15000, to_date('06-11-99', 'dd-mm-yy'),'House Number 1, Barddhaman, West Bengal')

1 row created.

SQL> insert into Chef values ('&staff_id_chef','&name',&salary, &date_of_birth,'&address');
Enter value for staff_id_chef: DP2CF
Enter value for name: Yuvraj Kumar
Enter value for salary: 18000
Enter value for date_of_birth: to_date('04-05-99', 'dd-mm-yy')
Enter value for address: House Number 50 Ramnagar, Gwalior, Madhya Pradesh
old   1: insert into Chef values ('&staff_id_chef','&name',&salary, &date_of_birth,'&address')
new   1: insert into Chef values ('DP2CF','Yuvraj Kumar',18000, to_date('04-05-99', 'dd-mm-yy'),'House Number 50 Ramnagar, Gwalior, Madhya Pradesh')

1 row created.

SQL> insert into Chef values('DP3MR', 'Aryan Srivastava', 10000,to_date('18-08-00','dd-mm-yy'),'House Number 100 , Noida, Uttar Pradesh');

1 row created.

SQL> insert into Chef values('DP4MR', 'Ishan Aggarwal', 15000,to_date('01-01-04','dd-mm-yy'),'House Number 150 , Faridabad, Haryana');

1 row created.

SQL> insert into Chef values('DP5MR', 'Anagh Chattopadhyay', 20000,to_date('08-10-00','dd-mm-yy'),'House no99, Varanasi, UP');

1 row created.

SQL> insert into Chef values('DP6MR', 'Sushant Raj', 13000,to_date('27-11-98','dd-mm-yy'),'House no10, Mumbai, Maharashtra');

1 row created.

SQL> insert into Chef values('DP7MR', 'Sameer Bansal', 10000,to_date('05-09-01','dd-mm-yy'),'House no15, Kolkata, West Bengal');

1 row created.
```

Final Table



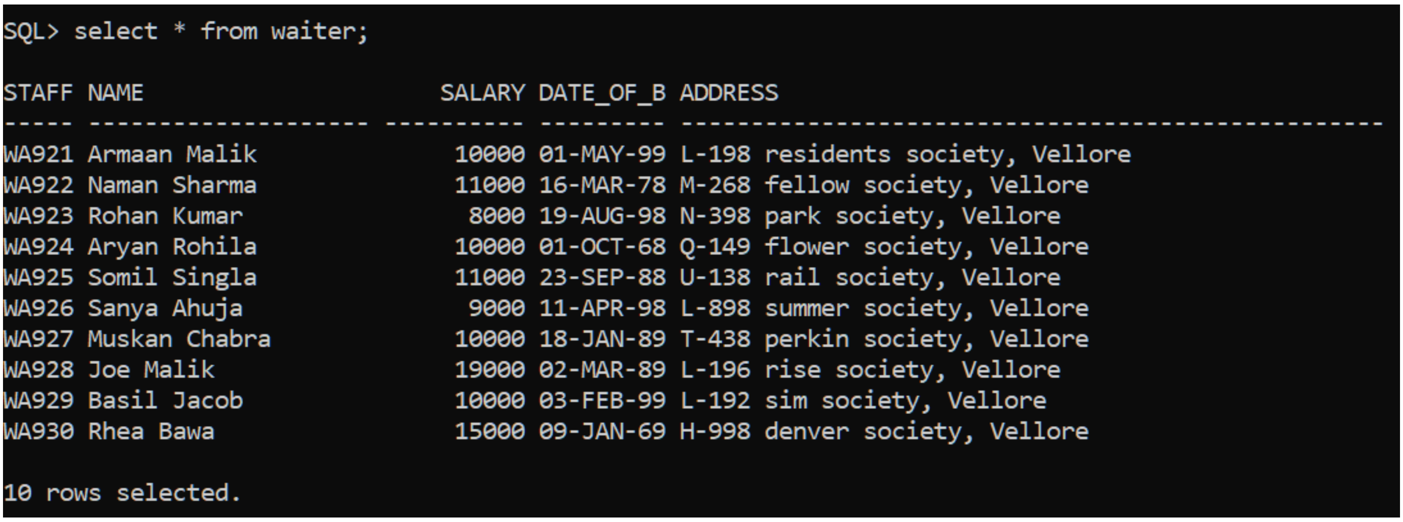
```
SQL> select * from chef;

STAFF NAME                     SALARY DATE_OF_B ADDRESS
----- -------------------- ---------- --------- --------------------------------------------------
DP1CF Rounak Basak              15000 06-NOV-99 House Number 1, Barddhaman, West Bengal
DP2CF Yuvraj Kumar              18000 04-MAY-99 House Number 50 Ramnagar, Gwalior, Madhya Pradesh
DP3CF Aryan Srivastava          10000 18-AUG-00 House Number 100 , Noida, Uttar Pradesh
DP4CF Ishan Aggarwal            15000 01-JAN-04 House Number 150 , Faridabad, Haryana
DP5CF Anagh Chattopadhyay       20000 08-OCT-00 House no99, Varanasi, UP
DP6CF Sushant Raj               13000 27-NOV-98 House no10, Mumbai, Maharashtra
DP7CF Sameer Bansal             10000 05-SEP-01 House no15, Kolkata, West Bengal

7 rows selected.
```

Waiter Table



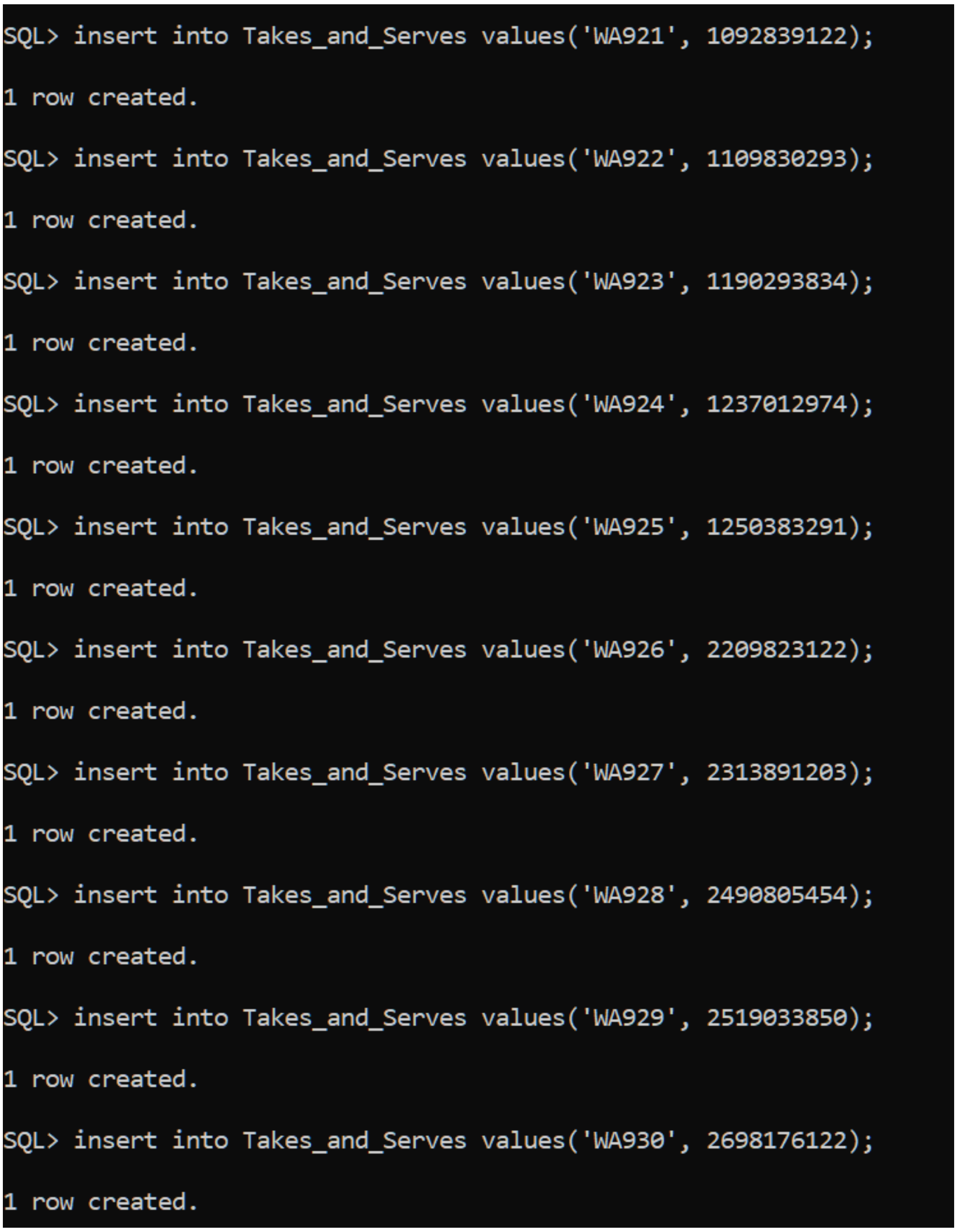
```
SQL> insert into waiter values('WA921','Armaan Malik',10000, to_date('01-05-1999','dd-mm-yy') ,'L-198 residents society, Vellore');

1 row created.

SQL> insert into waiter values('WA922','Naman Sharma', 11000, to_date('16-03-1978','dd-mm-yy'),'M-268 fellow society, Vellore');

1 row created.

SQL> insert into waiter values('WA923','Rohan Kumar', 8000, to_date('19-08-1998','dd-mm-yy'),'N-398 park society, Vellore');

1 row created.

SQL> insert into waiter values('WA924','Aryan Rohila', 10000, to_date(’01-10-1968','dd-mm-yy'),'Q-149 flower society, Vellore');

1 row created.

SQL> insert into waiter values('WA925','Somil Singla', 11000, to_date('23-09-1988','dd-mm-yy'),'U-138 rail society, Vellore');

1 row created.

SQL> insert into waiter values('WA926','Sanya Ahuja', 9000, to_date('11-04-1998','dd-mm-yy'),'L-898 summer society, Vellore');

1 row created.

SQL> insert into waiter values('WA927','Muskan Chabra', 10000, to_date('18-01-1989','dd-mm-yy'),'T-438 perkin society, Vellore');

1 row created.
```

Final Table



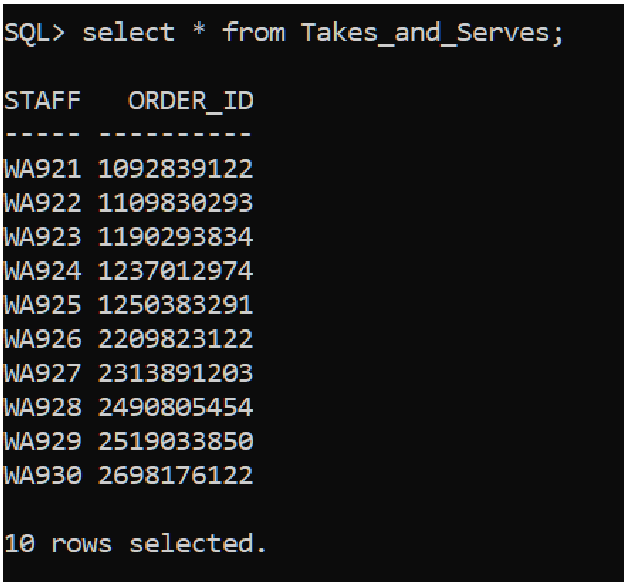
```
SQL> select * from waiter;

STAFF NAME                     SALARY DATE_OF_B ADDRESS
----- -------------------- ---------- --------- --------------------------------------------------
WA921 Armaan Malik              10000 01-MAY-99 L-198 residents society, Vellore
WA922 Naman Sharma              11000 16-MAR-78 M-268 fellow society, Vellore
WA923 Rohan Kumar                8000 19-AUG-98 N-398 park society, Vellore
WA924 Aryan Rohila              10000 01-OCT-68 Q-149 flower society, Vellore
WA925 Somil Singla              11000 23-SEP-88 U-138 rail society, Vellore
WA926 Sanya Ahuja                9000 11-APR-98 L-898 summer society, Vellore
WA927 Muskan Chabra             10000 18-JAN-89 T-438 perkin society, Vellore
WA928 Joe Malik                 19000 02-MAR-89 L-196 rise society, Vellore
WA929 Basil Jacob               10000 03-FEB-99 L-192 sim society, Vellore
WA930 Rhea Bawa                 15000 09-JAN-69 H-998 denver society, Vellore

10 rows selected.
```

Takes_and_Serves Table



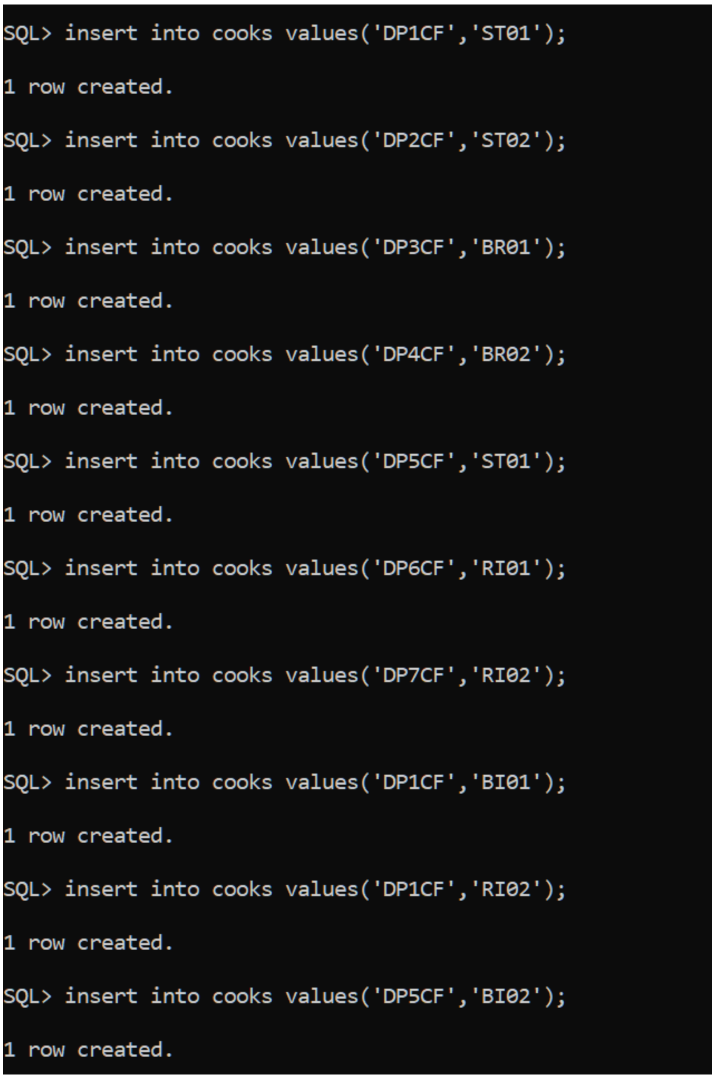
```
SQL> insert into Takes_and_Serves values('WA921', 1092839122);

1 row created.

SQL> insert into Takes_and_Serves values('WA922', 1109830293);

1 row created.

SQL> insert into Takes_and_Serves values('WA923', 1190293834);

1 row created.

SQL> insert into Takes_and_Serves values('WA924', 1237012974);

1 row created.

SQL> insert into Takes_and_Serves values('WA925', 1250383291);

1 row created.

SQL> insert into Takes_and_Serves values('WA926', 2209823122);

1 row created.

SQL> insert into Takes_and_Serves values('WA927', 2313891203);

1 row created.

SQL> insert into Takes_and_Serves values('WA928', 2490805454);

1 row created.

SQL> insert into Takes_and_Serves values('WA929', 2519033850);

1 row created.

SQL> insert into Takes_and_Serves values('WA930', 2698176122);

1 row created.
```

Final Table



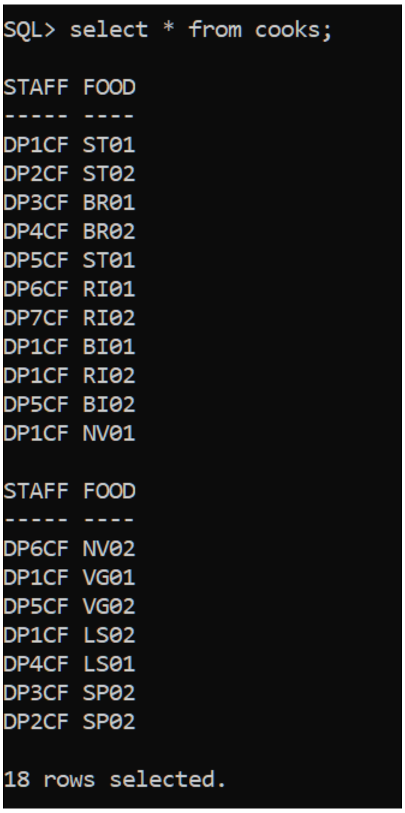
```
SQL> select * from Takes_and_Serves;

STAFF   ORDER_ID
----- ----------
WA921 1092839122
WA922 1109830293
WA923 1190293834
WA924 1237012974
WA925 1250383291
WA926 2209823122
WA927 2313891203
WA928 2490805454
WA929 2519033850
WA930 2698176122

10 rows selected.
```

Cooks Table



```
SQL> insert into cooks values('DP1CF','ST01');

1 row created.

SQL> insert into cooks values('DP2CF','ST02');

1 row created.

SQL> insert into cooks values('DP3CF','BR01');

1 row created.

SQL> insert into cooks values('DP4CF','BR02');

1 row created.

SQL> insert into cooks values('DP5CF','ST01');

1 row created.

SQL> insert into cooks values('DP6CF','RI01');

1 row created.

SQL> insert into cooks values('DP7CF','RI02');

1 row created.

SQL> insert into cooks values('DP1CF','BI01');

1 row created.

SQL> insert into cooks values('DP1CF','RI02');

1 row created.

SQL> insert into cooks values('DP5CF','BI02');

1 row created.
```

Final Table

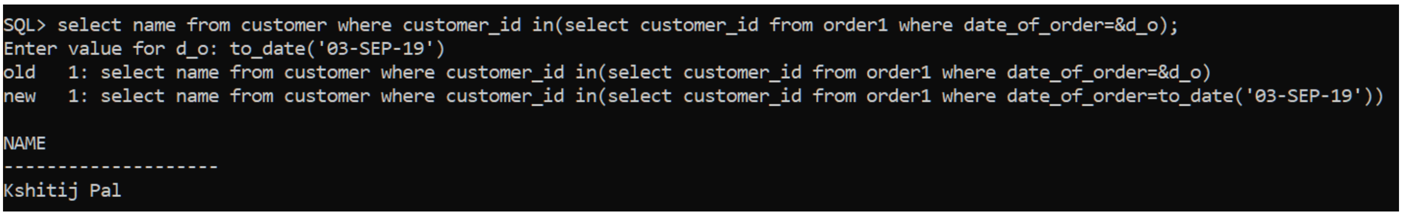


```
SQL> select * from cooks;

STAFF FOOD
----- ----
DP1CF ST01
DP2CF ST02
DP3CF BR01
DP4CF BR02
DP5CF ST01
DP6CF RI01
DP7CF RI02
DP1CF BI01
DP1CF RI02
DP5CF BI02
DP1CF NV01

STAFF FOOD
----- ----
DP6CF NV02
DP1CF VG01
DP5CF VG02
DP1CF LS02
DP4CF LS01
DP3CF SP02
DP2CF SP02

18 rows selected.
```

FUNCTIONAL REQUIREMENTS

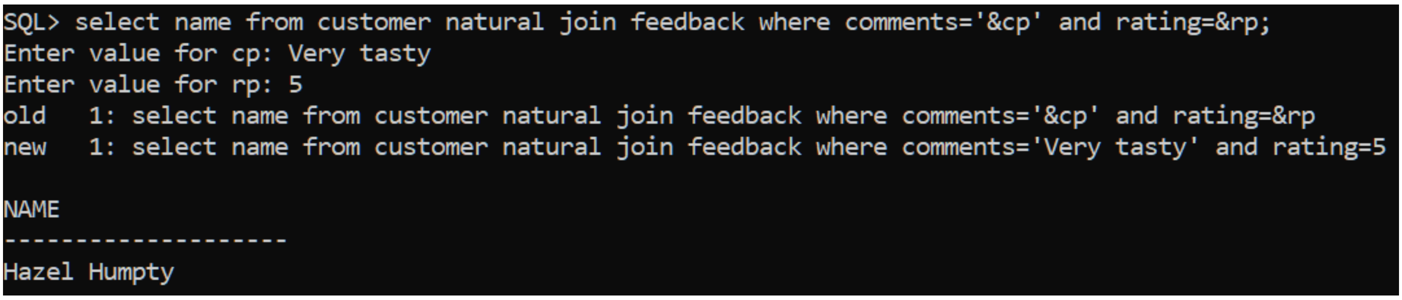
Data Retrieval

1) Name of the customer who gave order on the entered date



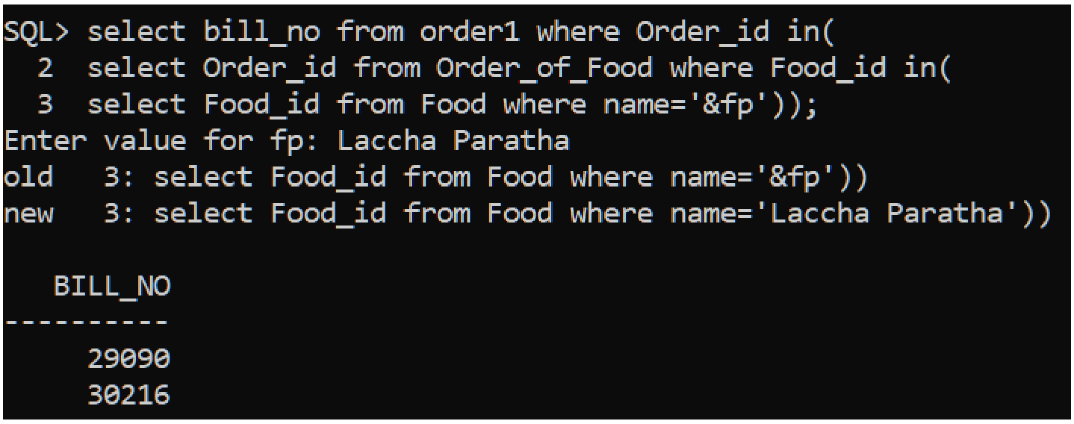
```
SQL> select name from customer where customer_id in(select customer_id from order1 where date_of_order=&d_o);
Enter value for d_o: to_date('03-SEP-19')
old   1: select name from customer where customer_id in(select customer_id from order1 where date_of_order=&d_o)
new   1: select name from customer where customer_id in(select customer_id from order1 where date_of_order=to_date('03-SEP-19'))

NAME
--------------------
Kshitij Pal
```

2) Name of the customer who did gave particular comment and rating



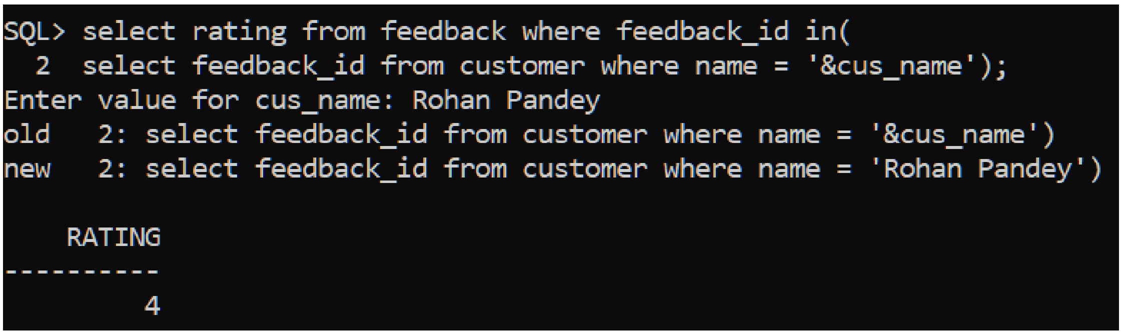
```
SQL> select name from customer natural join feedback where comments='&cp' and rating=&rp;
Enter value for cp: Very tasty
Enter value for rp: 5
old   1: select name from customer natural join feedback where comments='&cp' and rating=&rp
new   1: select name from customer natural join feedback where comments='Very tasty' and rating=5

NAME
--------------------
Hazel Humpty
```

3) All bill no for a particular food



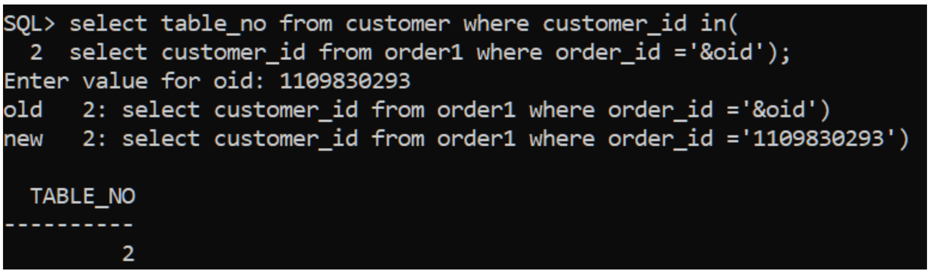
```
SQL> select bill_no from order1 where Order_id in(
  2  select Order_id from Order_of_Food where Food_id in(
  3  select Food_id from Food where name='&fp'));
Enter value for fp: Laccha Paratha
old   3: select Food_id from Food where name='&fp'))
new   3: select Food_id from Food where name='Laccha Paratha'))

   BILL_NO
----------
     29090
     30216
```

4) Rating given by a particular customer



```
SQL> select rating from feedback where feedback_id in(
  2  select feedback_id from customer where name = '&cus_name');
Enter value for cus_name: Rohan Pandey
old   2: select feedback_id from customer where name = '&cus_name')
new   2: select feedback_id from customer where name = 'Rohan Pandey')

    RATING
----------
         4
```

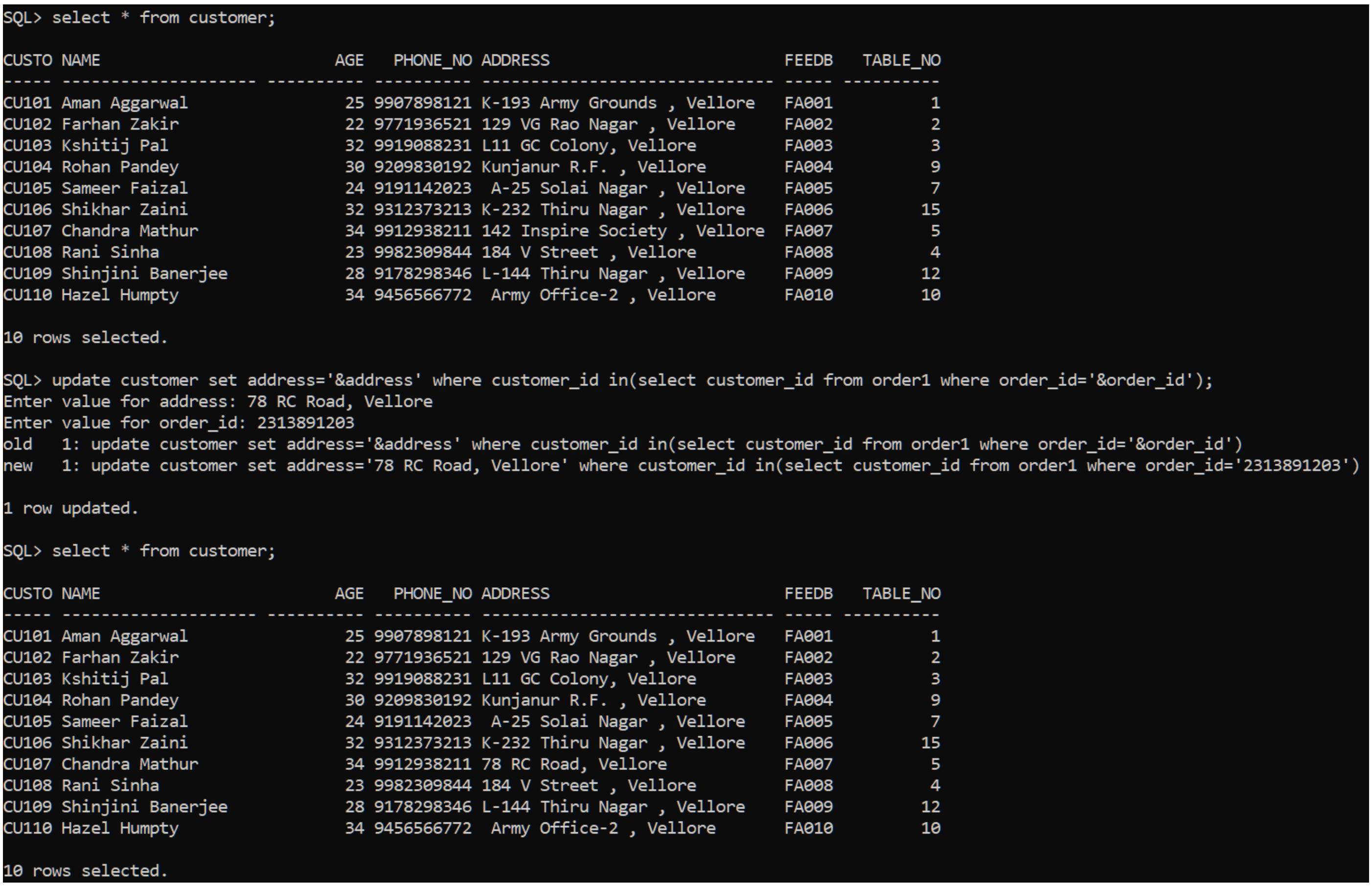
5) Table_no where a particular order was served



```
SQL> select table_no from customer where customer_id in(
  2  select customer_id from order1 where order_id ='&oid');
Enter value for oid: 1109830293
old   2: select customer_id from order1 where order_id ='&oid')
new   2: select customer_id from order1 where order_id ='1109830293')

  TABLE_NO
----------
         2
```

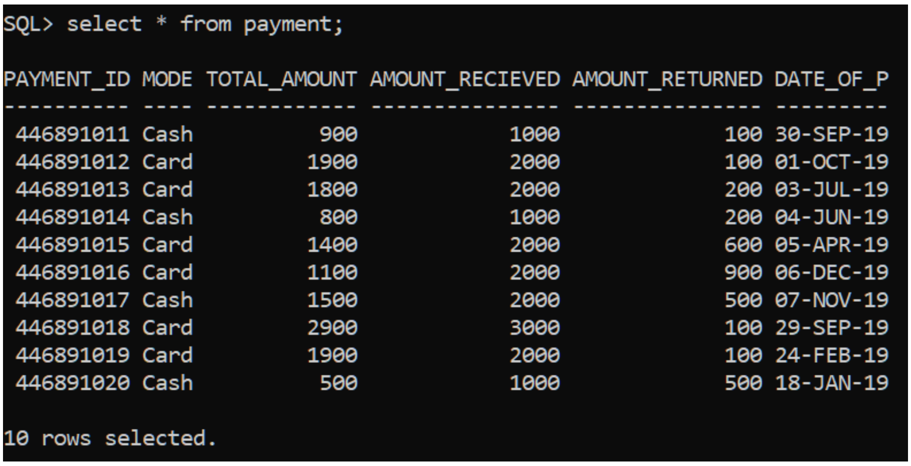
Modification of data

1) A customer has changed his address. So, his address needs to be updated for delivery of his order by order_id.

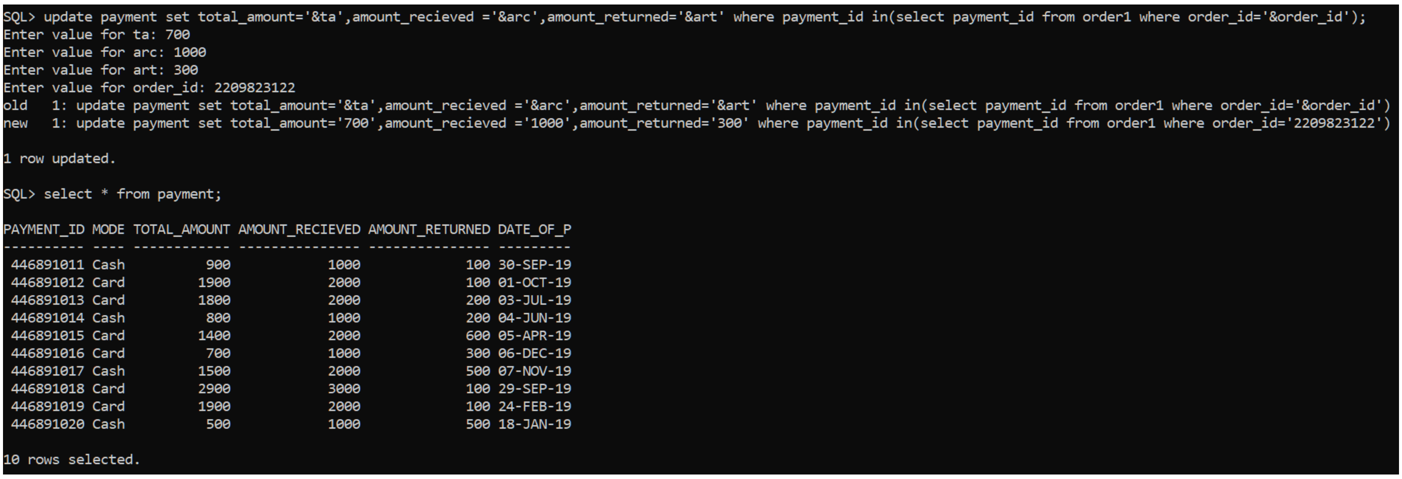


```
SQL> select * from customer;

CUSTO NAME                       AGE   PHONE_NO ADDRESS                        FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ------------------------------ ----- ----------
CU101 Aman Aggarwal               25 9907898121 K-193 Army Grounds , Vellore   FA001          1
CU102 Farhan Zakir                22 9771936521 129 VG Rao Nagar , Vellore     FA002          2
CU103 Kshitij Pal                 32 9919088231 L11 GC Colony, Vellore         FA003          3
CU104 Rohan Pandey                30 9209830192 Kunjanur R.F. , Vellore        FA004          9
CU105 Sameer Faizal               24 9191142023  A-25 Solai Nagar , Vellore    FA005          7
CU106 Shikhar Zaini               32 9312373213 K-232 Thiru Nagar , Vellore    FA006         15
CU107 Chandra Mathur              34 9912938211 142 Inspire Society , Vellore  FA007          5
CU108 Rani Sinha                  23 9982309844 184 V Street , Vellore         FA008          4
CU109 Shinjini Banerjee           28 9178298346 L-144 Thiru Nagar , Vellore    FA009         12
CU110 Hazel Humpty                34 9456566772  Army Office-2 , Vellore       FA010         10

10 rows selected.

SQL> update customer set address='&address' where customer_id in(select customer_id from order1 where order_id='&order_id');
Enter value for address: 78 RC Road, Vellore
Enter value for order_id: 2313891203
old   1: update customer set address='&address' where customer_id in(select customer_id from order1 where order_id='&order_id')
new   1: update customer set address='78 RC Road, Vellore' where customer_id in(select customer_id from order1 where order_id='2313891203')

1 row updated.

SQL> select * from customer;

CUSTO NAME                       AGE   PHONE_NO ADDRESS                        FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ------------------------------ ----- ----------
CU101 Aman Aggarwal               25 9907898121 K-193 Army Grounds , Vellore   FA001          1
CU102 Farhan Zakir                22 9771936521 129 VG Rao Nagar , Vellore     FA002          2
CU103 Kshitij Pal                 32 9919088231 L11 GC Colony, Vellore         FA003          3
CU104 Rohan Pandey                30 9209830192 Kunjanur R.F. , Vellore        FA004          9
CU105 Sameer Faizal               24 9191142023  A-25 Solai Nagar , Vellore    FA005          7
CU106 Shikhar Zaini               32 9312373213 K-232 Thiru Nagar , Vellore    FA006         15
CU107 Chandra Mathur              34 9912938211 78 RC Road, Vellore            FA007          5
CU108 Rani Sinha                  23 9982309844 184 V Street , Vellore         FA008          4
CU109 Shinjini Banerjee           28 9178298346 L-144 Thiru Nagar , Vellore    FA009         12
CU110 Hazel Humpty                34 9456566772  Army Office-2 , Vellore       FA010         10

10 rows selected.
```

2) The amount of payment needs to be updated for an order

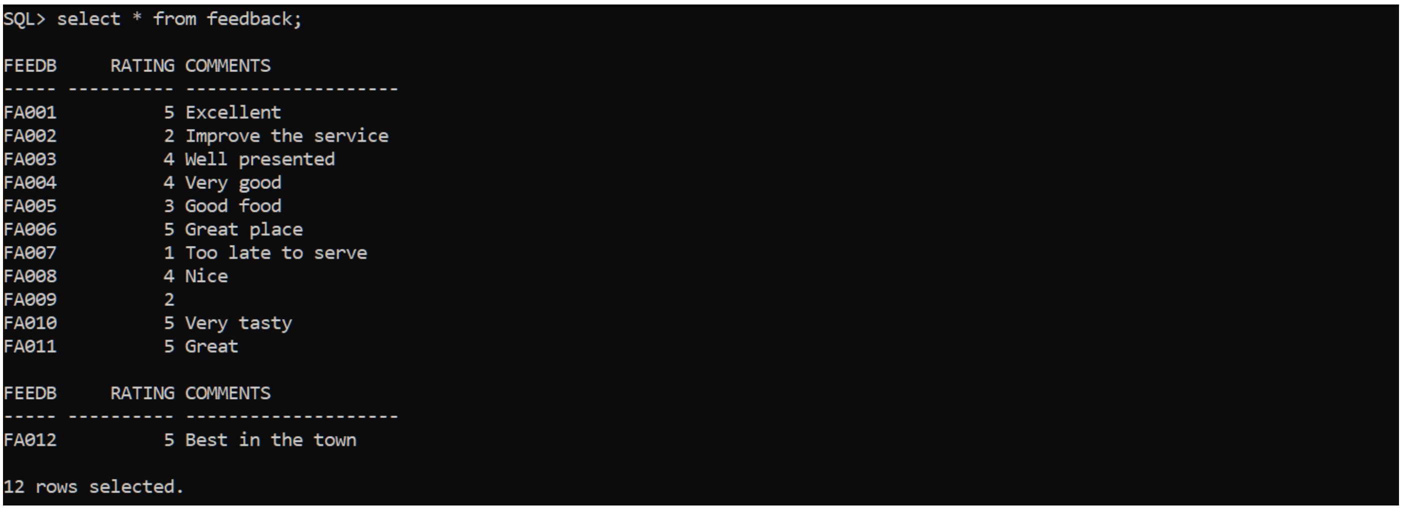


```
SQL> select * from payment;

PAYMENT_ID MODE TOTAL_AMOUNT AMOUNT_RECIEVED AMOUNT_RETURNED DATE_OF_P
---------- ---- ------------ --------------- --------------- ---------
 446891011 Cash          900            1000             100 30-SEP-19
 446891012 Card         1900            2000             100 01-OCT-19
 446891013 Card         1800            2000             200 03-JUL-19
 446891014 Cash          800            1000             200 04-JUN-19
 446891015 Card         1400            2000             600 05-APR-19
 446891016 Card         1100            2000             900 06-DEC-19
 446891017 Cash         1500            2000             500 07-NOV-19
 446891018 Card         2900            3000             100 29-SEP-19
 446891019 Card         1900            2000             100 24-FEB-19
 446891020 Cash          500            1000             500 18-JAN-19

10 rows selected.
```

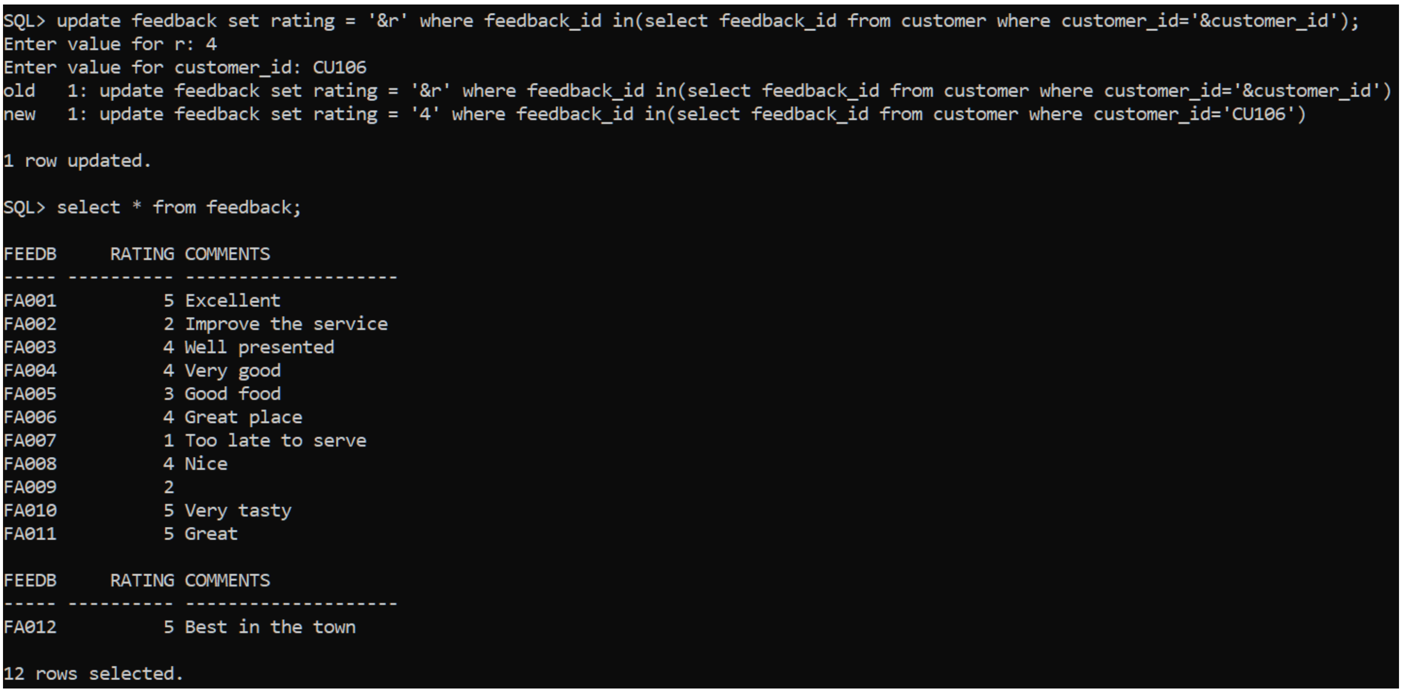


```
SQL> update payment set total_amount='&ta',amount_recieved ='&arc',amount_returned='&art' where payment_id in(select payment_id from order1 where order_id='&order_id');
Enter value for ta: 700
Enter value for arc: 1000
Enter value for art: 300
Enter value for order_id: 2209823122
old   1: update payment set total_amount='&ta',amount_recieved ='&arc',amount_returned='&art' where payment_id in(select payment_id from order1 where order_id='&order_id')
new   1: update payment set total_amount='700',amount_recieved ='1000',amount_returned='300' where payment_id in(select payment_id from order1 where order_id='2209823122')

1 row updated.

SQL> select * from payment;

PAYMENT_ID MODE TOTAL_AMOUNT AMOUNT_RECIEVED AMOUNT_RETURNED DATE_OF_P
---------- ---- ------------ --------------- --------------- ---------
 446891011 Cash          900            1000             100 30-SEP-19
 446891012 Card         1900            2000             100 01-OCT-19
 446891013 Card         1800            2000             200 03-JUL-19
 446891014 Cash          800            1000             200 04-JUN-19
 446891015 Card         1400            2000             600 05-APR-19
 446891016 Card          700            1000             300 06-DEC-19
 446891017 Cash         1500            2000             500 07-NOV-19
 446891018 Card         2900            3000             100 29-SEP-19
 446891019 Card         1900            2000             100 24-FEB-19
 446891020 Cash          500            1000             500 18-JAN-19

10 rows selected.
```

3) A customer decides to change his feedback for services of the restaurant

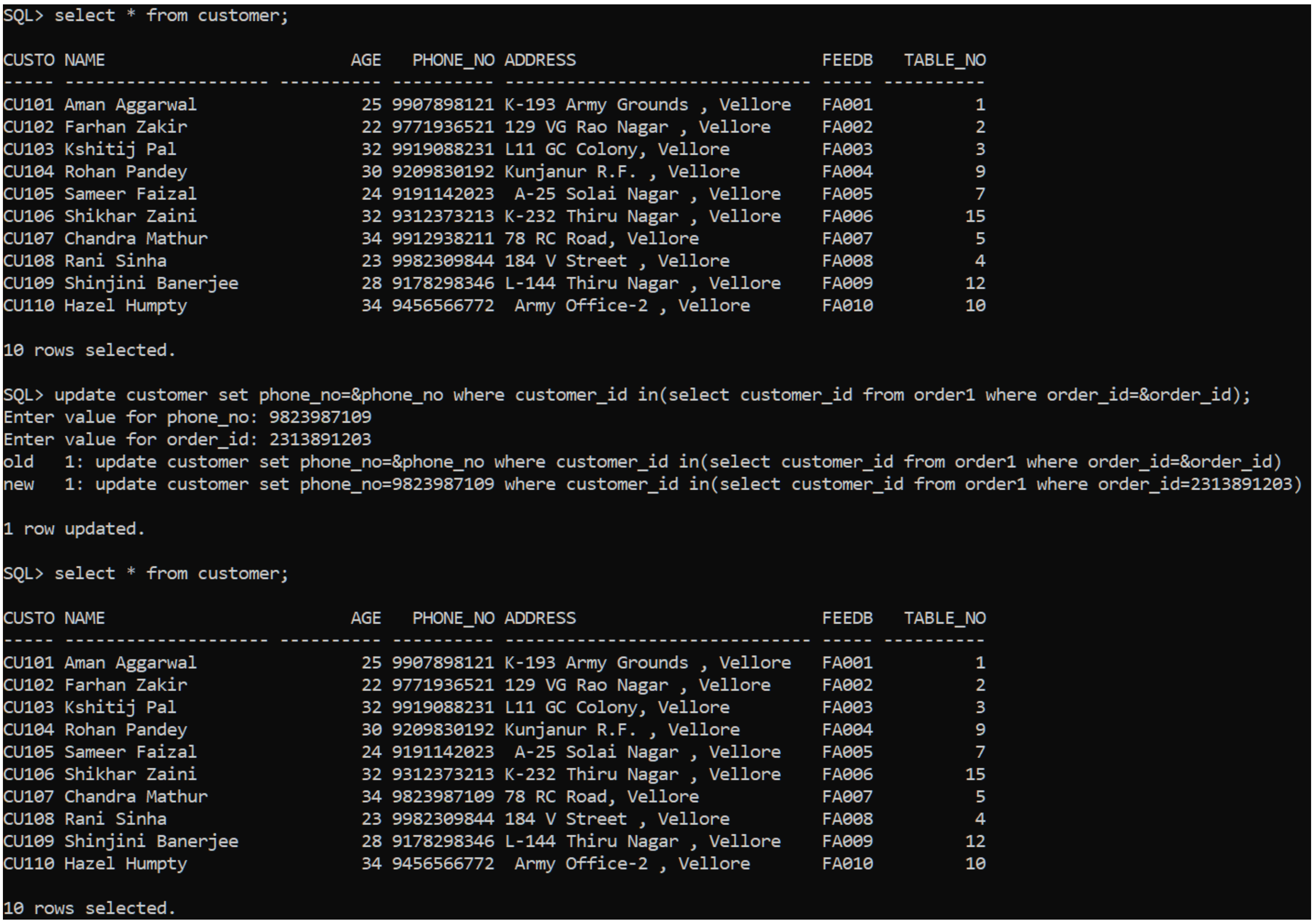


```
SQL> select * from feedback;

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA001          5 Excellent
FA002          2 Improve the service
FA003          4 Well presented
FA004          4 Very good
FA005          3 Good food
FA006          5 Great place
FA007          1 Too late to serve
FA008          4 Nice
FA009          2
FA010          5 Very tasty
FA011          5 Great

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA012          5 Best in the town

12 rows selected.
```



```
SQL> update feedback set rating = '&r' where feedback_id in(select feedback_id from customer where customer_id='&customer_id');
Enter value for r: 4
Enter value for customer_id: CU106
old   1: update feedback set rating = '&r' where feedback_id in(select feedback_id from customer where customer_id='&customer_id')
new   1: update feedback set rating = '4' where feedback_id in(select feedback_id from customer where customer_id='CU106')

1 row updated.

SQL> select * from feedback;

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA001          5 Excellent
FA002          2 Improve the service
FA003          4 Well presented
FA004          4 Very good
FA005          3 Good food
FA006          4 Great place
FA007          1 Too late to serve
FA008          4 Nice
FA009          2
FA010          5 Very tasty
FA011          5 Great

FEEDB     RATING COMMENTS
----- ---------- --------------------
FA012          5 Best in the town

12 rows selected.
```

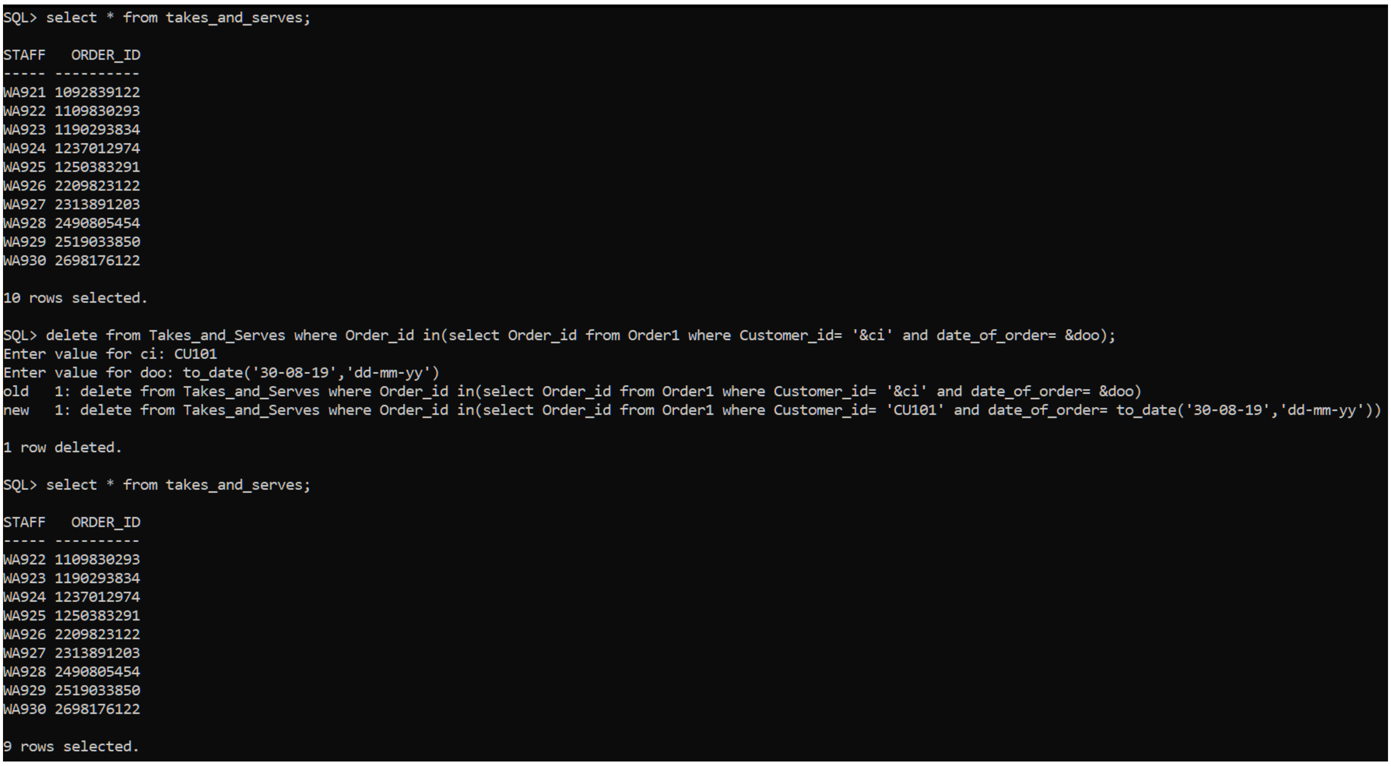
4) A customer decides to change his mobile number at time of delivery of an order



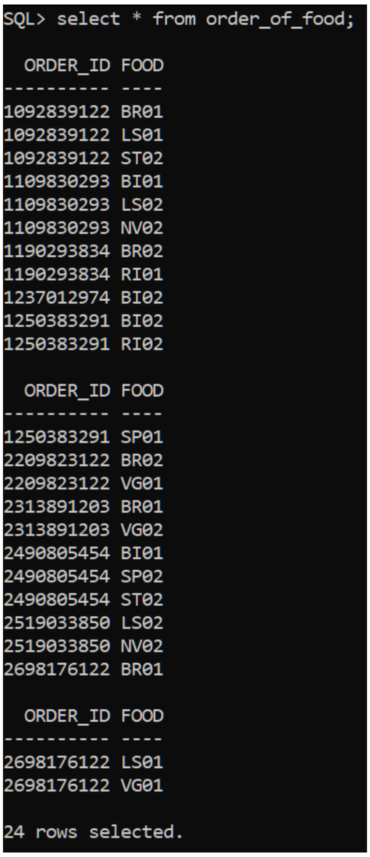
```
SQL> select * from customer;

CUSTO NAME                        AGE   PHONE_NO ADDRESS                      FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ---------------------------- ----- ----------
CU101 Aman Aggarwal                25 9907898121 K-193 Army Grounds , Vellore  FA001          1
CU102 Farhan Zakir                 22 9771936521 129 VG Rao Nagar , Vellore    FA002          2
CU103 Kshitij Pal                  32 9919088231 L11 GC Colony, Vellore        FA003          3
CU104 Rohan Pandey                 30 9209830192 Kunjanur R.F. , Vellore       FA004          9
CU105 Sameer Faizal                24 9191142023  A-25 Solai Nagar , Vellore   FA005          7
CU106 Shikhar Zaini                32 9312373213 K-232 Thiru Nagar , Vellore   FA006         15
CU107 Chandra Mathur               34 9912938211 78 RC Road, Vellore           FA007          5
CU108 Rani Sinha                   23 9982309844 184 V Street , Vellore        FA008          4
CU109 Shinjini Banerjee            28 9178298346 L-144 Thiru Nagar , Vellore   FA009         12
CU110 Hazel Humpty                 34 9456566772  Army Office-2 , Vellore      FA010         10

10 rows selected.

SQL> update customer set phone_no=&phone_no where customer_id in(select customer_id from order1 where order_id=&order_id);
Enter value for phone_no: 9823987109
Enter value for order_id: 2313891203
old   1: update customer set phone_no=&phone_no where customer_id in(select customer_id from order1 where order_id=&order_id)
new   1: update customer set phone_no=9823987109 where customer_id in(select customer_id from order1 where order_id=2313891203)

1 row updated.

SQL> select * from customer;

CUSTO NAME                        AGE   PHONE_NO ADDRESS                      FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ---------------------------- ----- ----------
CU101 Aman Aggarwal                25 9907898121 K-193 Army Grounds , Vellore  FA001          1
CU102 Farhan Zakir                 22 9771936521 129 VG Rao Nagar , Vellore    FA002          2
CU103 Kshitij Pal                  32 9919088231 L11 GC Colony, Vellore        FA003          3
CU104 Rohan Pandey                 30 9209830192 Kunjanur R.F. , Vellore       FA004          9
CU105 Sameer Faizal                24 9191142023  A-25 Solai Nagar , Vellore   FA005          7
CU106 Shikhar Zaini                32 9312373213 K-232 Thiru Nagar , Vellore   FA006         15
CU107 Chandra Mathur               34 9823987109 78 RC Road, Vellore           FA007          5
CU108 Rani Sinha                   23 9982309844 184 V Street , Vellore        FA008          4
CU109 Shinjini Banerjee            28 9178298346 L-144 Thiru Nagar , Vellore   FA009         12
CU110 Hazel Humpty                 34 9456566772  Army Office-2 , Vellore      FA010         10

10 rows selected.
```

Removal of old data

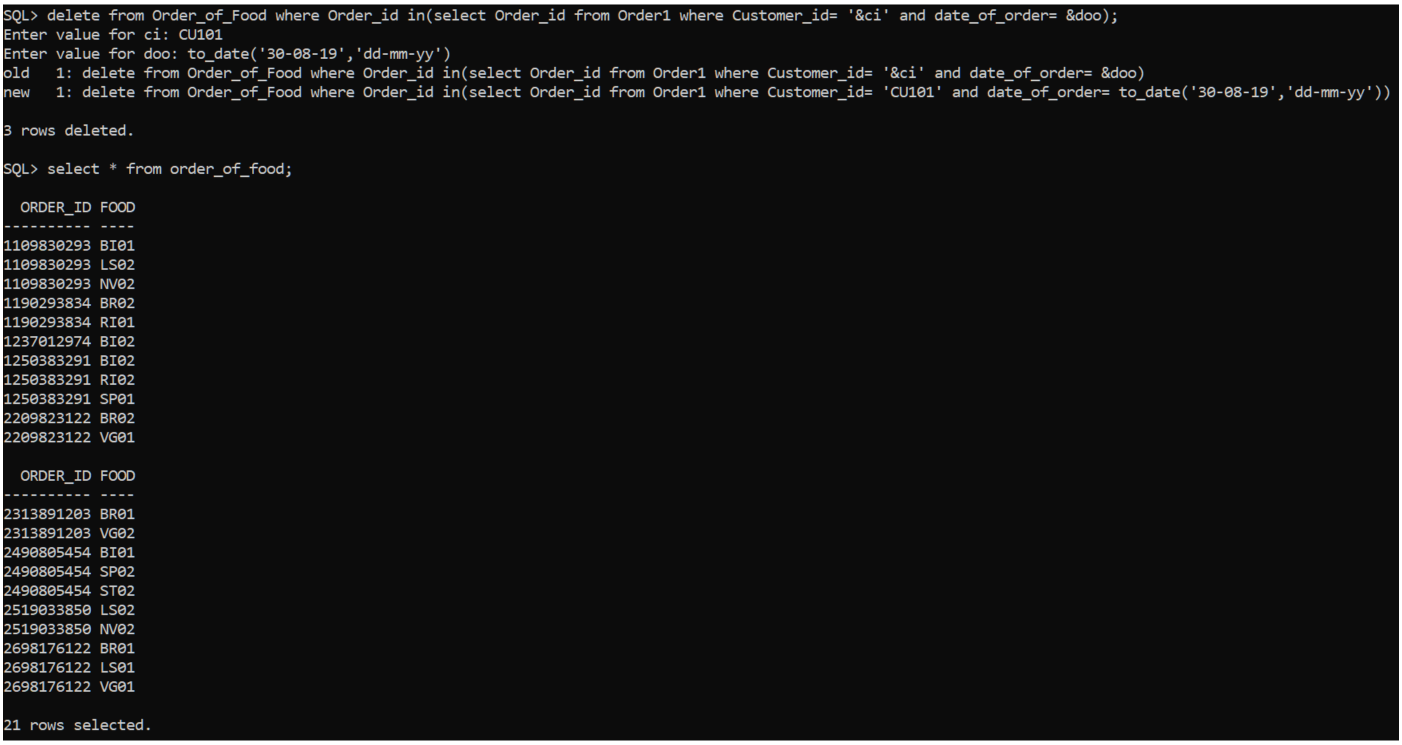
1) A customer decides to delete his order at a particular date. So by his id and date of order, the staff id , food id and order id should be deleted.



```
SQL> select * from takes_and_serves;

STAFF   ORDER_ID
----- ----------
WA921 1092839122
WA922 1109830293
WA923 1190293834
WA924 1237012974
WA925 1250383291
WA926 2209823122
WA927 2313891203
WA928 2490805454
WA929 2519033850
WA930 2698176122

10 rows selected.

SQL> delete from Takes_and_Serves where Order_id in(select Order_id from Order1 where Customer_id= '&ci' and date_of_order= &doo);
Enter value for ci: CU101
Enter value for doo: to_date('30-08-19','dd-mm-yy')
old   1: delete from Takes_and_Serves where Order_id in(select Order_id from Order1 where Customer_id= '&ci' and date_of_order= &doo)
new   1: delete from Takes_and_Serves where Order_id in(select Order_id from Order1 where Customer_id= 'CU101' and date_of_order= to_date('30-08-19','dd-mm-yy'))

1 row deleted.

SQL> select * from takes_and_serves;

STAFF   ORDER_ID
----- ----------
WA922 1109830293
WA923 1190293834
WA924 1237012974
WA925 1250383291
WA926 2209823122
WA927 2313891203
WA928 2490805454
WA929 2519033850
WA930 2698176122

9 rows selected.
```

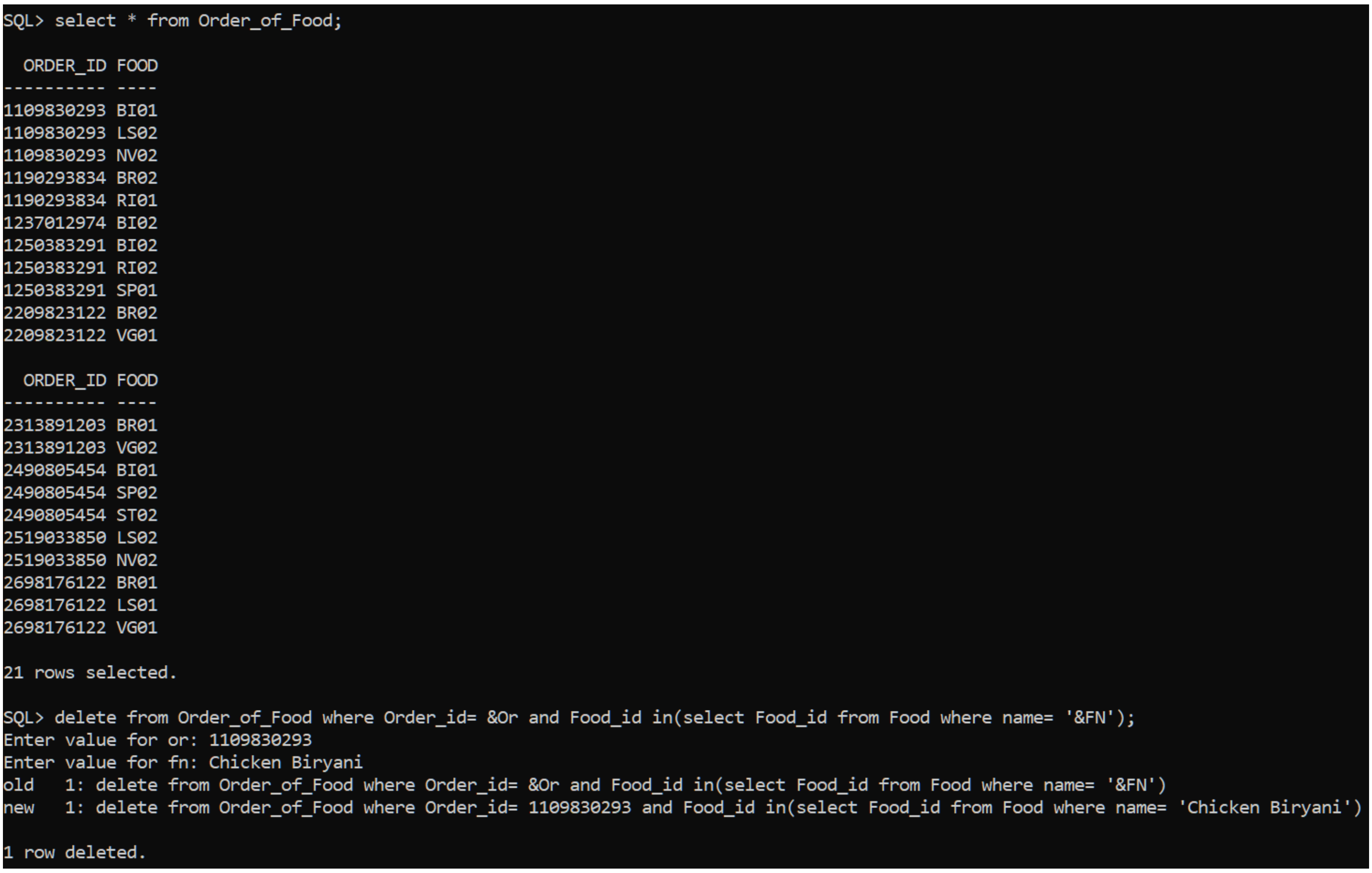


```
SQL> select * from order_of_food;

  ORDER_ID FOOD
---------- ----
1092839122 BR01
1092839122 LS01
1092839122 ST02
1109830293 BI01
1109830293 LS02
1109830293 NV02
1190293834 BR02
1190293834 RI01
1237012974 BI02
1250383291 BI02
1250383291 RI02

  ORDER_ID FOOD
---------- ----
1250383291 SP01
2209823122 BR02
2209823122 VG01
2313891203 BR01
2313891203 VG02
2490805454 BI01
2490805454 SP02
2490805454 ST02
2519033850 LS02
2519033850 NV02
2698176122 BR01

  ORDER_ID FOOD
---------- ----
2698176122 LS01
2698176122 VG01

24 rows selected.
```

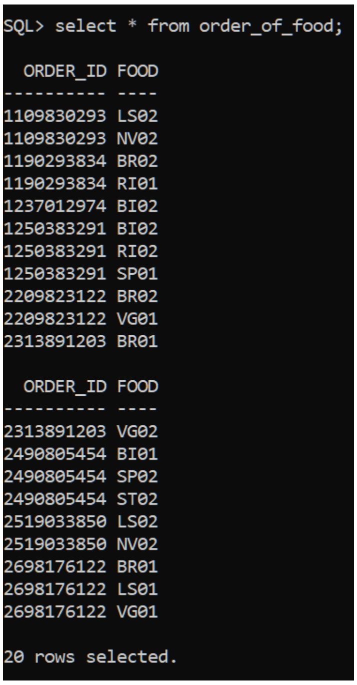


```
SQL> delete from Order_of_Food where Order_id in(select Order_id from Order1 where Customer_id= '&ci' and date_of_order= &doo);
Enter value for ci: CU101
Enter value for doo: to_date('30-08-19','dd-mm-yy')
old   1: delete from Order_of_Food where Order_id in(select Order_id from Order1 where Customer_id= '&ci' and date_of_order= &doo)
new   1: delete from Order_of_Food where Order_id in(select Order_id from Order1 where Customer_id= 'CU101' and date_of_order= to_date('30-08-19','dd-mm-yy'))

3 rows deleted.

SQL> select * from order_of_food;

  ORDER_ID FOOD
---------- ----
1109830293 BI01
1109830293 LS02
1109830293 NV02
1190293834 BR02
1190293834 RI01
1237012974 BI02
1250383291 BI02
1250383291 RI02
1250383291 SP01
2209823122 BR02
2209823122 VG01

  ORDER_ID FOOD
---------- ----
2313891203 BR01
2313891203 VG02
2490805454 BI01
2490805454 SP02
2490805454 ST02
2519033850 LS02
2519033850 NV02
2698176122 BR01
2698176122 LS01
2698176122 VG01

21 rows selected.
```

2) A customer decides to delete a food item from his order. So, its food id needs to be deleted.

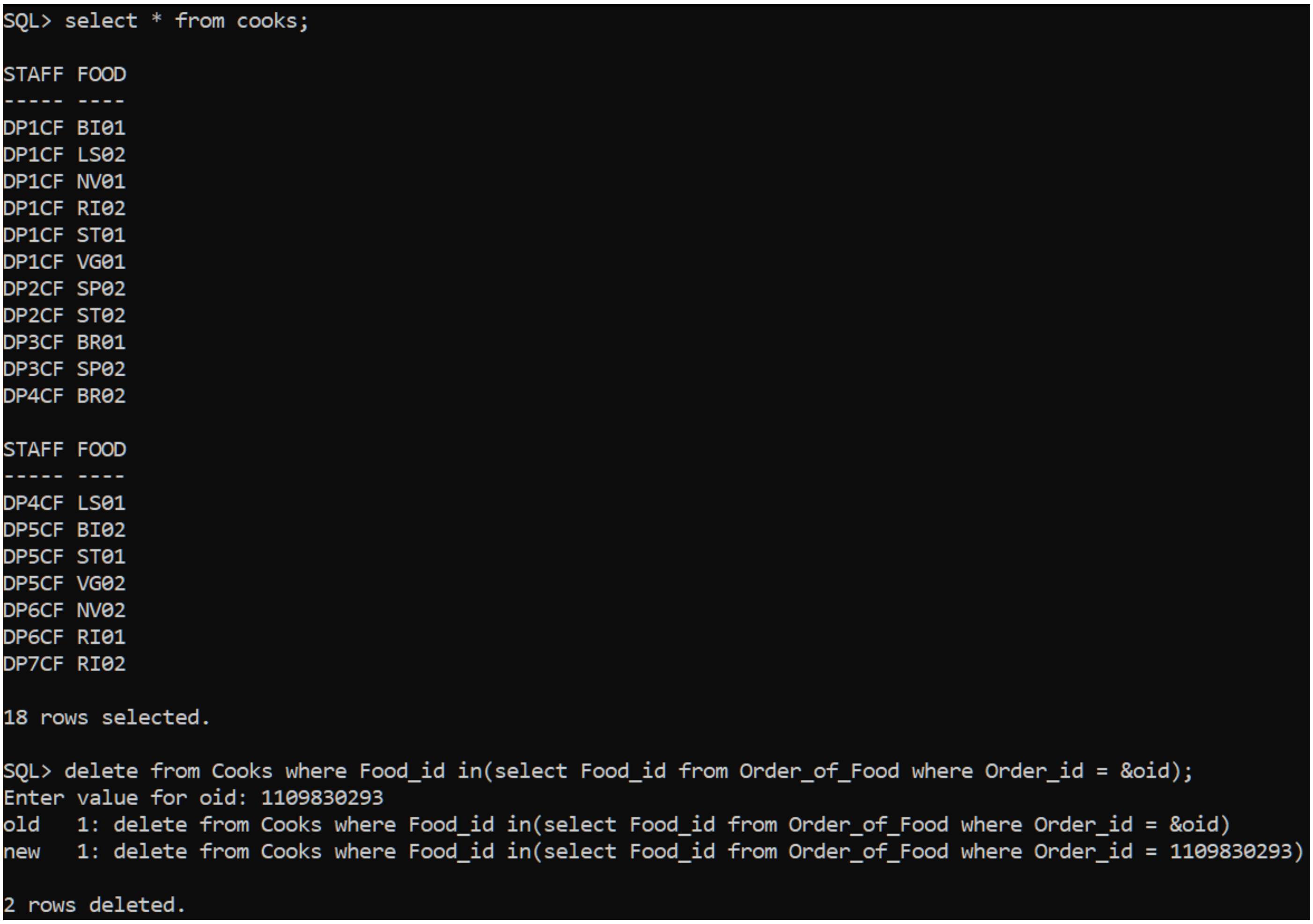


```
SQL> select * from Order_of_Food;

  ORDER_ID FOOD
---------- ----
1109830293 BI01
1109830293 LS02
1109830293 NV02
1190293834 BR02
1190293834 RI01
1237012974 BI02
1250383291 BI02
1250383291 RI02
1250383291 SP01
2209823122 BR02
2209823122 VG01

  ORDER_ID FOOD
---------- ----
2313891203 BR01
2313891203 VG02
2490805454 BI01
2490805454 SP02
2490805454 ST02
2519033850 LS02
2519033850 NV02
2698176122 BR01
2698176122 LS01
2698176122 VG01

21 rows selected.

SQL> delete from Order_of_Food where Order_id= &Or and Food_id in(select Food_id from Food where name= '&FN');
Enter value for or: 1109830293
Enter value for fn: Chicken Biryani
old   1: delete from Order_of_Food where Order_id= &Or and Food_id in(select Food_id from Food where name= '&FN')
new   1: delete from Order_of_Food where Order_id= 1109830293 and Food_id in(select Food_id from Food where name= 'Chicken Biryani')

1 row deleted.
```

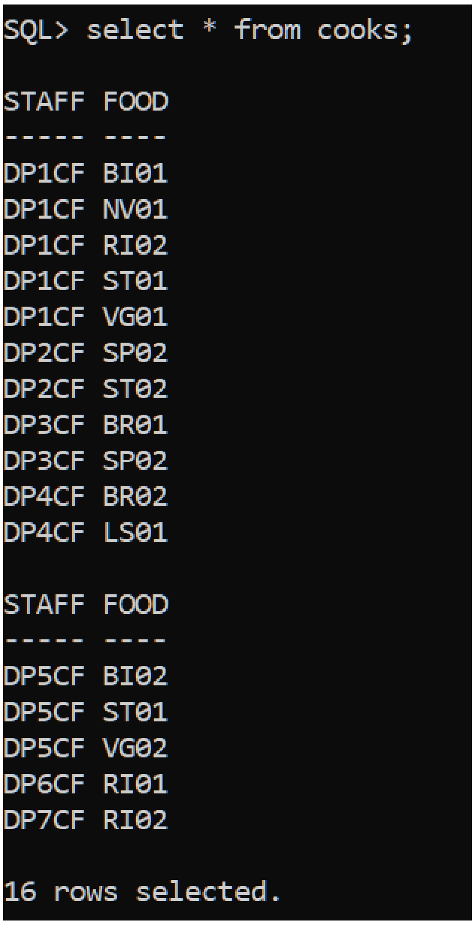


```
SQL> select * from order_of_food;

  ORDER_ID FOOD
---------- ----
1109830293 LS02
1109830293 NV02
1190293834 BR02
1190293834 RI01
1237012974 BI02
1250383291 BI02
1250383291 RI02
1250383291 SP01
2209823122 BR02
2209823122 VG01
2313891203 BR01

  ORDER_ID FOOD
---------- ----
2313891203 VG02
2490805454 BI01
2490805454 SP02
2490805454 ST02
2519033850 LS02
2519033850 NV02
2698176122 BR01
2698176122 LS01
2698176122 VG01

20 rows selected.
```

3) An order has been placed to cook which gets rejected at last moment. So its food id for cook needs to be deleted.



```
SQL> select * from cooks;

STAFF FOOD
----- ----
DP1CF BI01
DP1CF LS02
DP1CF NV01
DP1CF RI02
DP1CF ST01
DP1CF VG01
DP2CF SP02
DP2CF ST02
DP3CF BR01
DP3CF SP02
DP4CF BR02

STAFF FOOD
----- ----
DP4CF LS01
DP5CF BI02
DP5CF ST01
DP5CF VG02
DP6CF NV02
DP6CF RI01
DP7CF RI02

18 rows selected.

SQL> delete from Cooks where Food_id in(select Food_id from Order_of_Food where Order_id = &oid);
Enter value for oid: 1109830293
old   1: delete from Cooks where Food_id in(select Food_id from Order_of_Food where Order_id = &oid)
new   1: delete from Cooks where Food_id in(select Food_id from Order_of_Food where Order_id = 1109830293)

2 rows deleted.
```

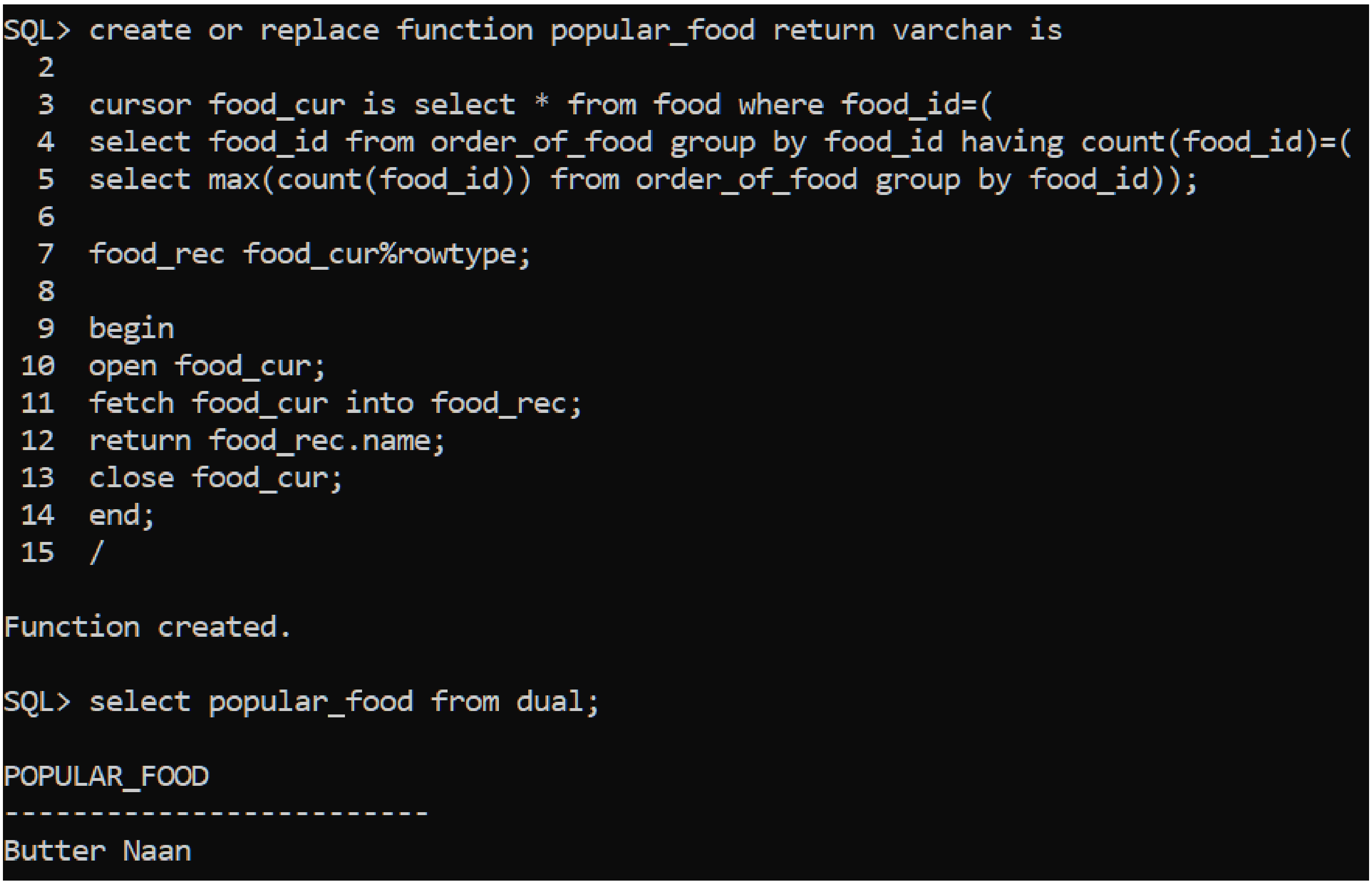


```
SQL> select * from cooks;

STAFF FOOD
----- ----
DP1CF BI01
DP1CF NV01
DP1CF RI02
DP1CF ST01
DP1CF VG01
DP2CF SP02
DP2CF ST02
DP3CF BR01
DP3CF SP02
DP4CF BR02
DP4CF LS01

STAFF FOOD
----- ----
DP5CF BI02
DP5CF ST01
DP5CF VG02
DP6CF RI01
DP7CF RI02

16 rows selected.
```

PL/SQL

Function

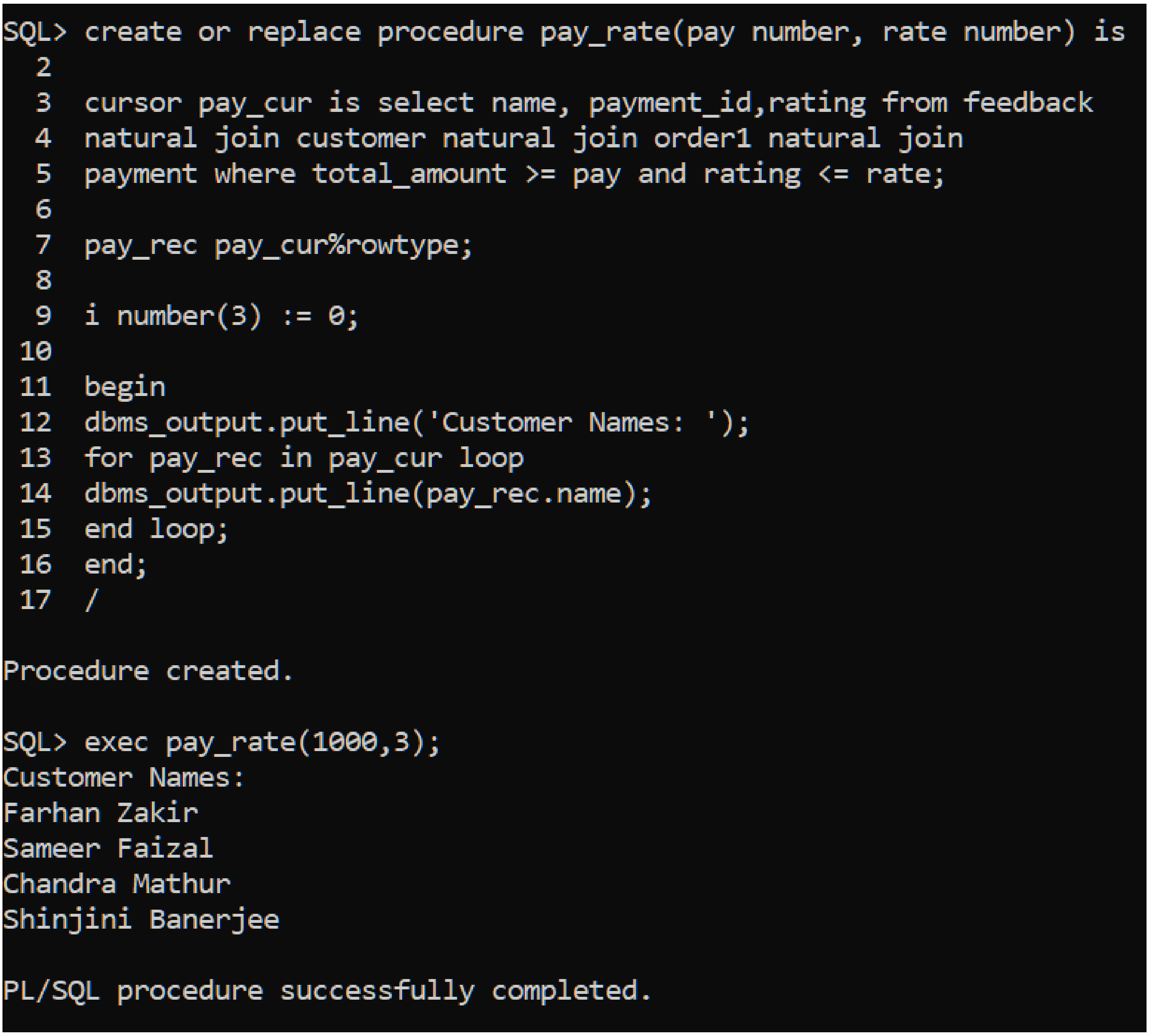
This function can be used to obtain the most popular food in the restaurant.



```
SQL> create or replace function popular_food return varchar is

cursor food_cur is select * from food where food_id=(
select food_id from order_of_food group by food_id having count(food_id)=(
select max(count(food_id)) from order_of_food group by food_id));

food_rec food_cur%rowtype;

begin
open food_cur;
fetch food_cur into food_rec;
return food_rec.name;
close food_cur;
end;
/

Function created.

SQL> select popular_food from dual;

POPULAR_FOOD
-------------------------
Butter Naan
```

Procedure

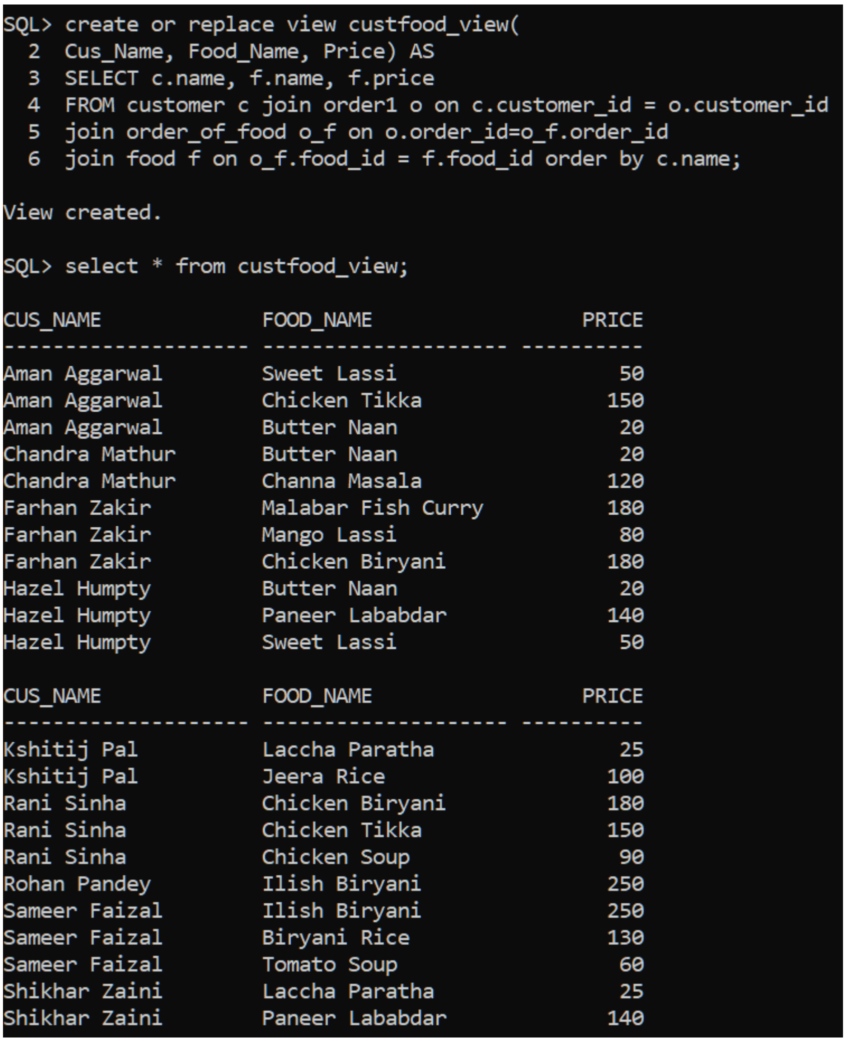
This procedure can be used to view the list of the customers who have paid greater than a particular amount and given rating below a particular rating.



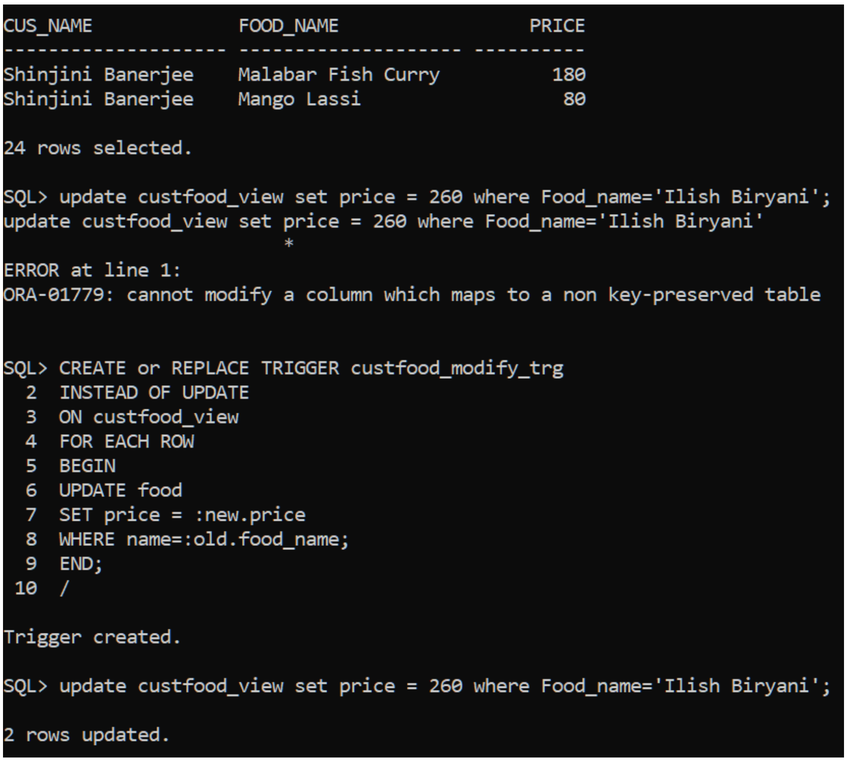
```
SQL> create or replace procedure pay_rate(pay number, rate number) is

     cursor pay_cur is select name, payment_id,rating from feedback
     natural join customer natural join order1 natural join
     payment where total_amount >= pay and rating <= rate;

     pay_rec pay_cur%rowtype;

     i number(3) := 0;

     begin
     dbms_output.put_line('Customer Names: ');
     for pay_rec in pay_cur loop
     dbms_output.put_line(pay_rec.name);
     end loop;
     end;
     /

Procedure created.

SQL> exec pay_rate(1000,3);
Customer Names:
Farhan Zakir
Sameer Faizal
Chandra Mathur
Shinjini Banerjee

PL/SQL procedure successfully completed.
```

Triggers

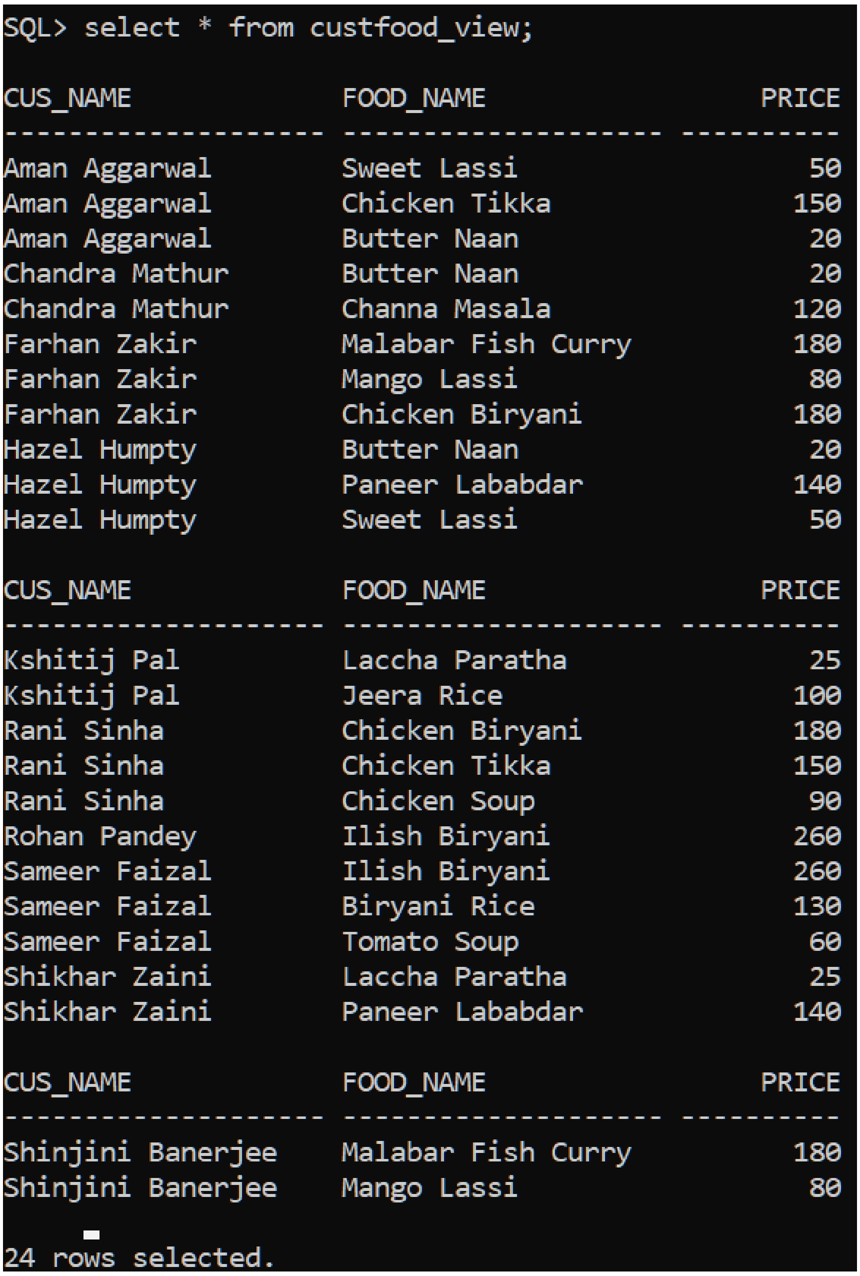
1) Instead of Trigger – it helps to update the value of price in the custfood_view directly



```
SQL> create or replace view custfood_view(
     Cus_Name, Food_Name, Price) AS
     SELECT c.name, f.name, f.price
     FROM customer c join order1 o on c.customer_id = o.customer_id
     join order_of_food o_f on o.order_id=o_f.order_id
     join food f on o_f.food_id = f.food_id order by c.name;

View created.

SQL> select * from custfood_view;

CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Aman Aggarwal        Sweet Lassi                  50
Aman Aggarwal        Chicken Tikka               150
Aman Aggarwal        Butter Naan                  20
Chandra Mathur       Butter Naan                  20
Chandra Mathur       Channa Masala               120
Farhan Zakir         Malabar Fish Curry          180
Farhan Zakir         Mango Lassi                  80
Farhan Zakir         Chicken Biryani             180
Hazel Humpty         Butter Naan                  20
Hazel Humpty         Paneer Lababdar             140
Hazel Humpty         Sweet Lassi                  50

CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Kshitij Pal          Laccha Paratha               25
Kshitij Pal          Jeera Rice                  100
Rani Sinha           Chicken Biryani             180
Rani Sinha           Chicken Tikka               150
Rani Sinha           Chicken Soup                 90
Rohan Pandey         Ilish Biryani               250
Sameer Faizal        Ilish Biryani               250
Sameer Faizal        Biryani Rice                130
Sameer Faizal        Tomato Soup                  60
Shikhar Zaini        Laccha Paratha               25
Shikhar Zaini        Paneer Lababdar             140
```

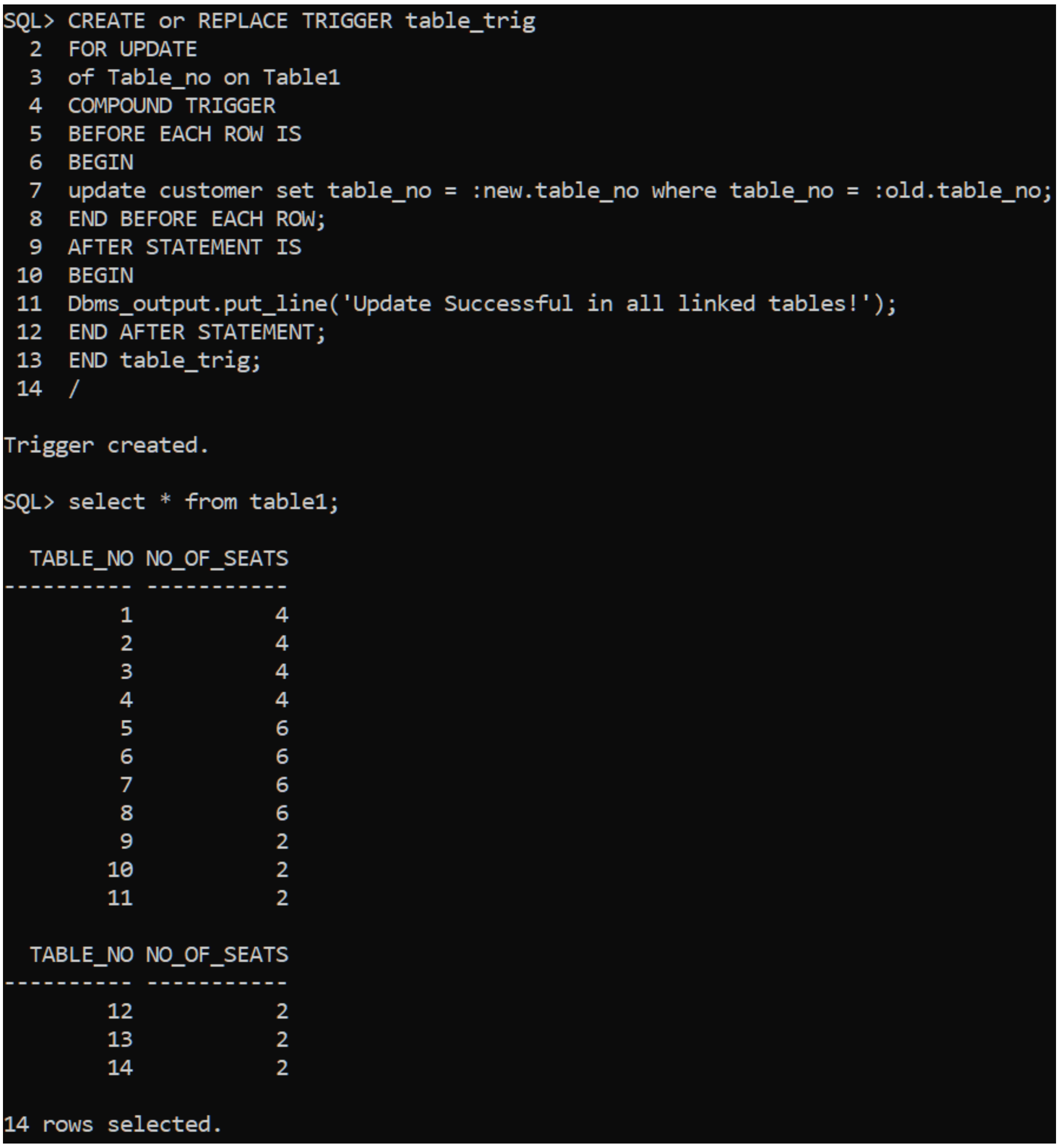


```
CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Shinjini Banerjee    Malabar Fish Curry          180
Shinjini Banerjee    Mango Lassi                  80

24 rows selected.

SQL> update custfood_view set price = 260 where Food_name='Ilish Biryani';
update custfood_view set price = 260 where Food_name='Ilish Biryani'
                         *
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table


SQL> CREATE or REPLACE TRIGGER custfood_modify_trg
     INSTEAD OF UPDATE
     ON custfood_view
     FOR EACH ROW
     BEGIN
     UPDATE food
     SET price = :new.price
     WHERE name=:old.food_name;
     END;
     /

Trigger created.

SQL> update custfood_view set price = 260 where Food_name='Ilish Biryani';

2 rows updated.
```

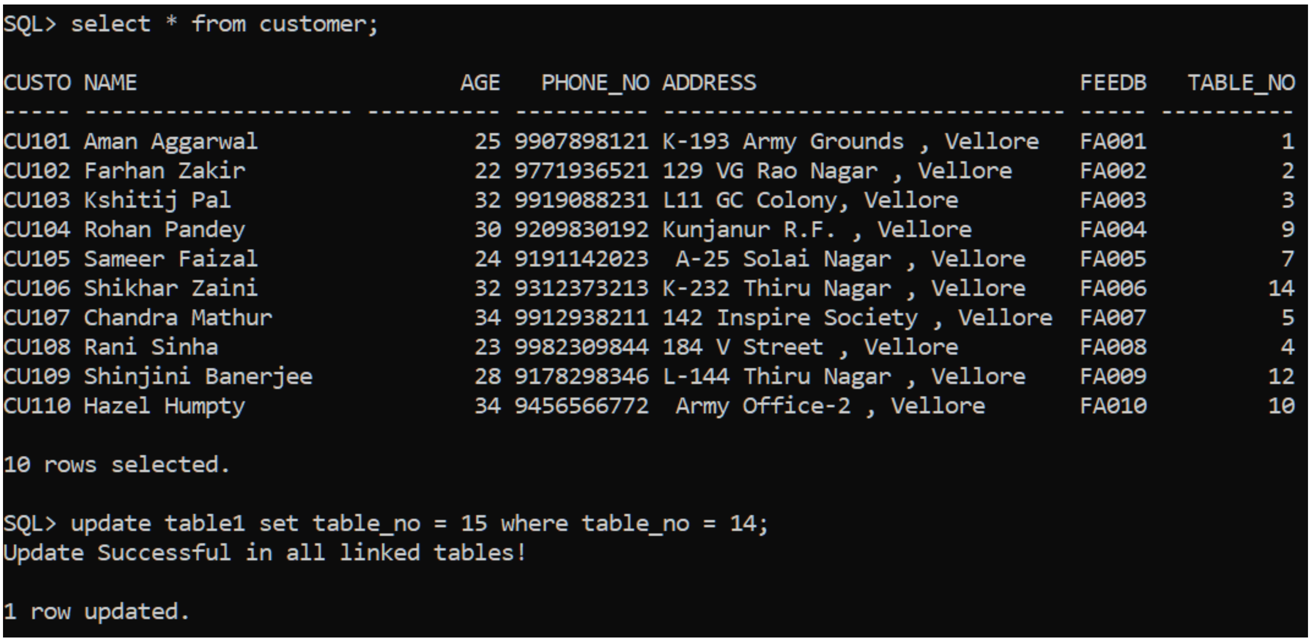


```
SQL> select * from custfood_view;

CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Aman Aggarwal        Sweet Lassi                  50
Aman Aggarwal        Chicken Tikka               150
Aman Aggarwal        Butter Naan                  20
Chandra Mathur       Butter Naan                  20
Chandra Mathur       Channa Masala               120
Farhan Zakir         Malabar Fish Curry          180
Farhan Zakir         Mango Lassi                  80
Farhan Zakir         Chicken Biryani             180
Hazel Humpty         Butter Naan                  20
Hazel Humpty         Paneer Lababdar             140
Hazel Humpty         Sweet Lassi                  50

CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Kshitij Pal          Laccha Paratha               25
Kshitij Pal          Jeera Rice                  100
Rani Sinha           Chicken Biryani             180
Rani Sinha           Chicken Tikka               150
Rani Sinha           Chicken Soup                 90
Rohan Pandey         Ilish Biryani               260
Sameer Faizal        Ilish Biryani               260
Sameer Faizal        Biryani Rice                130
Sameer Faizal        Tomato Soup                  60
Shikhar Zaini        Laccha Paratha               25
Shikhar Zaini        Paneer Lababdar             140

CUS_NAME             FOOD_NAME                 PRICE
-------------------- -------------------- ----------
Shinjini Banerjee    Malabar Fish Curry          180
Shinjini Banerjee    Mango Lassi                  80

24 rows selected.
```

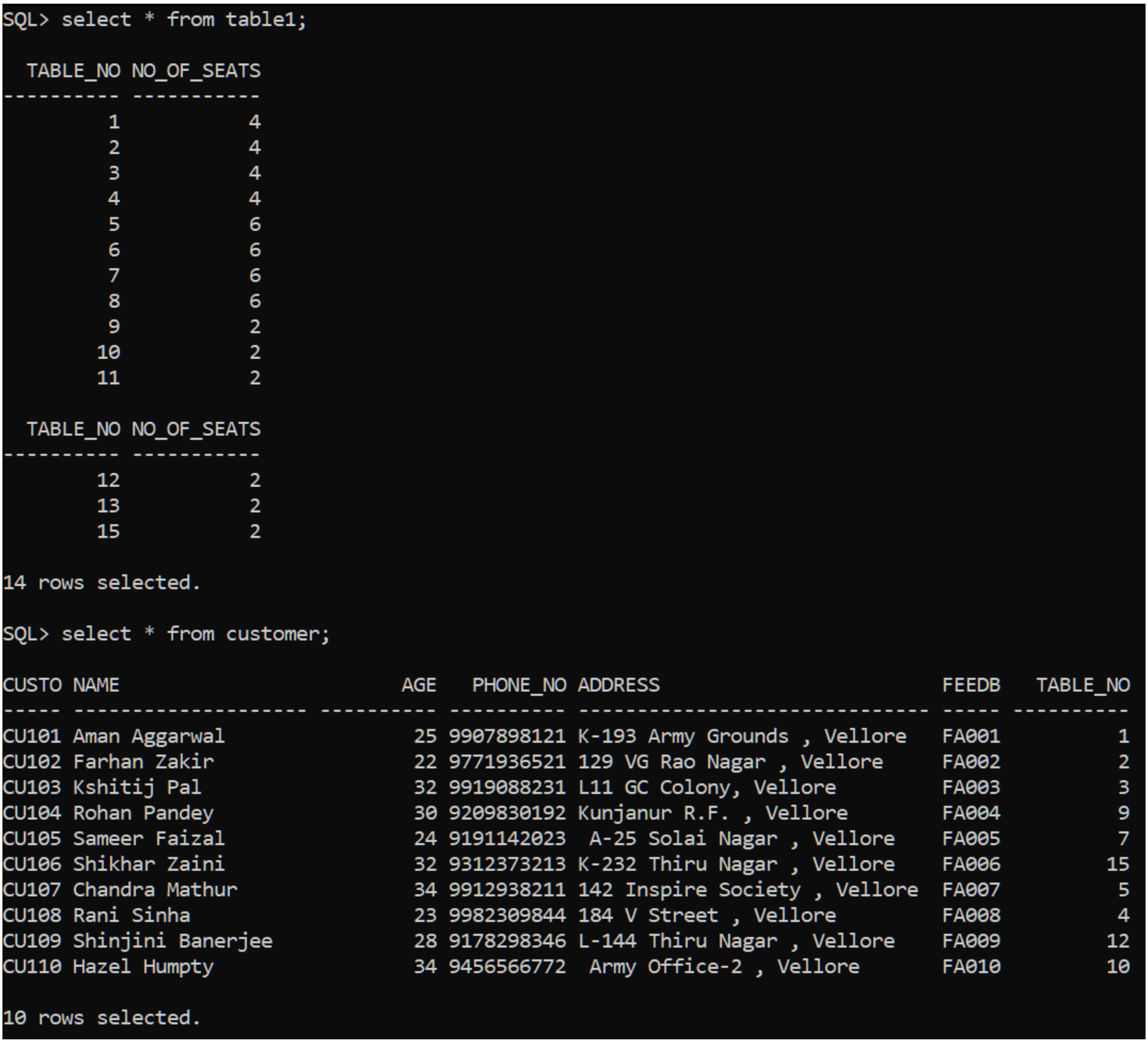
2) Compound Trigger - This trigger updates the data of table no in customer table automatically before the value gets updated in Table1 table and finally displays message after the successful update of both.



```
SQL> CREATE or REPLACE TRIGGER table_trig
     FOR UPDATE
     of Table_no on Table1
     COMPOUND TRIGGER
     BEFORE EACH ROW IS
     BEGIN
     update customer set table_no = :new.table_no where table_no = :old.table_no;
     END BEFORE EACH ROW;
     AFTER STATEMENT IS
     BEGIN
     Dbms_output.put_line('Update Successful in all linked tables!');
     END AFTER STATEMENT;
     END table_trig;
     /

Trigger created.

SQL> select * from table1;

  TABLE_NO NO_OF_SEATS
---------- -----------
         1           4
         2           4
         3           4
         4           4
         5           6
         6           6
         7           6
         8           6
         9           2
        10           2
        11           2

  TABLE_NO NO_OF_SEATS
---------- -----------
        12           2
        13           2
        14           2

14 rows selected.
```



```
SQL> select * from customer;

CUSTO NAME                        AGE   PHONE_NO ADDRESS                        FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ------------------------------ ----- ----------
CU101 Aman Aggarwal                25 9907898121 K-193 Army Grounds , Vellore   FA001          1
CU102 Farhan Zakir                 22 9771936521 129 VG Rao Nagar , Vellore     FA002          2
CU103 Kshitij Pal                  32 9919088231 L11 GC Colony, Vellore         FA003          3
CU104 Rohan Pandey                 30 9209830192 Kunjanur R.F. , Vellore        FA004          9
CU105 Sameer Faizal                24 9191142023  A-25 Solai Nagar , Vellore    FA005          7
CU106 Shikhar Zaini                32 9312373213 K-232 Thiru Nagar , Vellore    FA006         14
CU107 Chandra Mathur               34 9912938211 142 Inspire Society , Vellore  FA007          5
CU108 Rani Sinha                   23 9982309844 184 V Street , Vellore         FA008          4
CU109 Shinjini Banerjee            28 9178298346 L-144 Thiru Nagar , Vellore    FA009         12
CU110 Hazel Humpty                 34 9456566772  Army Office-2 , Vellore       FA010         10

10 rows selected.

SQL> update table1 set table_no = 15 where table_no = 14;
Update Successful in all linked tables!

1 row updated.
```



```
SQL> select * from table1;

  TABLE_NO NO_OF_SEATS
---------- -----------
         1           4
         2           4
         3           4
         4           4
         5           6
         6           6
         7           6
         8           6
         9           2
        10           2
        11           2

  TABLE_NO NO_OF_SEATS
---------- -----------
        12           2
        13           2
        15           2

14 rows selected.

SQL> select * from customer;

CUSTO NAME                        AGE   PHONE_NO ADDRESS                        FEEDB   TABLE_NO
----- -------------------- ---------- ---------- ------------------------------ ----- ----------
CU101 Aman Aggarwal                25 9907898121 K-193 Army Grounds , Vellore   FA001          1
CU102 Farhan Zakir                 22 9771936521 129 VG Rao Nagar , Vellore     FA002          2
CU103 Kshitij Pal                  32 9919088231 L11 GC Colony, Vellore         FA003          3
CU104 Rohan Pandey                 30 9209830192 Kunjanur R.F. , Vellore        FA004          9
CU105 Sameer Faizal                24 9191142023  A-25 Solai Nagar , Vellore    FA005          7
CU106 Shikhar Zaini                32 9312373213 K-232 Thiru Nagar , Vellore    FA006         15
CU107 Chandra Mathur               34 9912938211 142 Inspire Society , Vellore  FA007          5
CU108 Rani Sinha                   23 9982309844 184 V Street , Vellore         FA008          4
CU109 Shinjini Banerjee            28 9178298346 L-144 Thiru Nagar , Vellore    FA009         12
CU110 Hazel Humpty                 34 9456566772  Army Office-2 , Vellore       FA010         10

10 rows selected.
```