

DataNutts IT Solutions

PROJECT REPORT ON

RESTAURANT MANAGEMENT SYSTEM
.....

PROJECT GUIDE

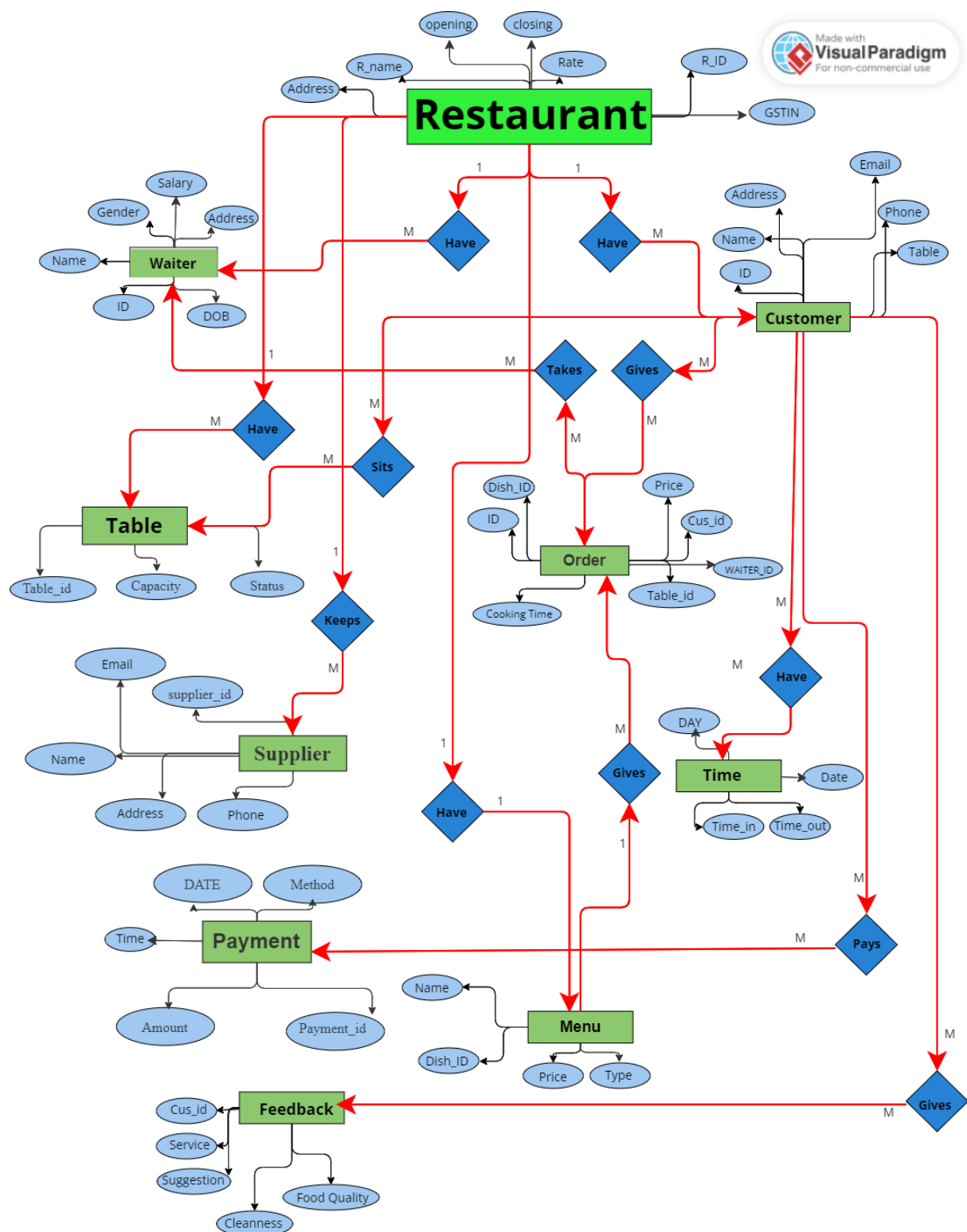
MAYANK KUKREJA

PREPARED AND SUBMITTED BY

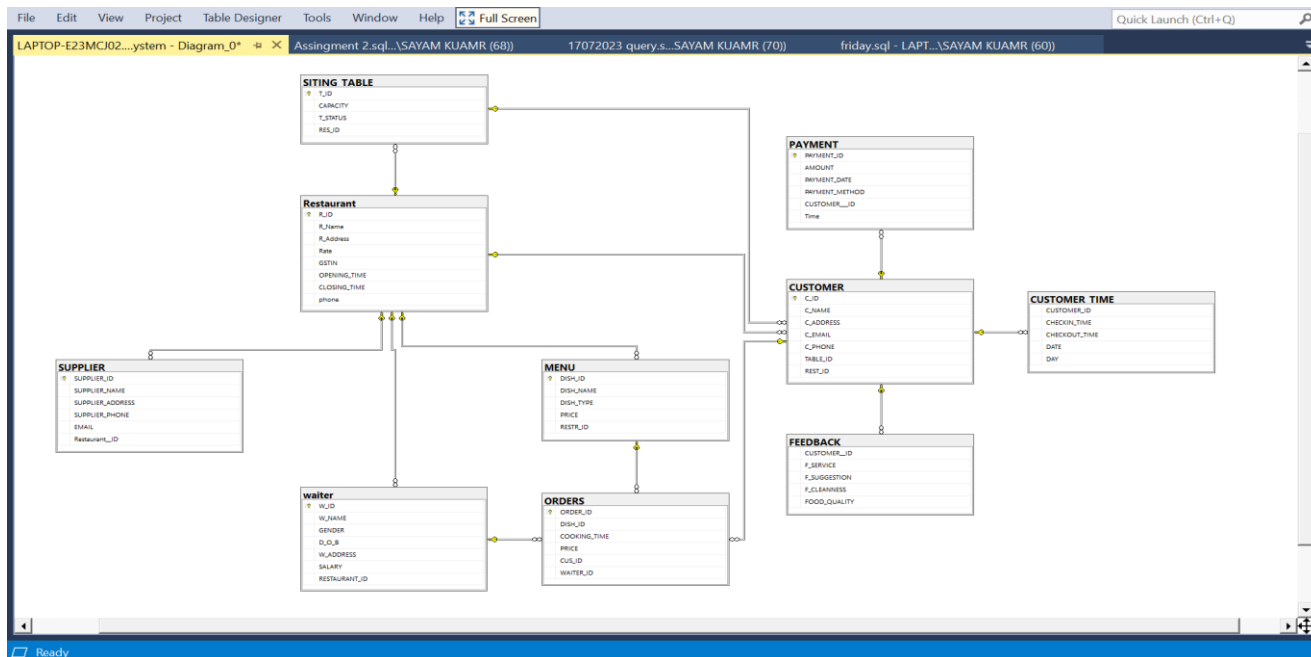
SATYAM KUMAR

(B.E(C.S.E))

ER DIAGRAM



SQL DIAGRAM



CREATION

---CREATED A DATABASE NAMED Restaurant_management_System---

create database Restaurant_management_System

---USING THE DATABASE---

use Restaurant_management_System

----HERE CREATED A TABLE NAMED RESTAURANT TO STORE RESTAURANT DATA----

create table Restaurant(

R_ID INT Primary key, /*----- creating R_ID as primary key for restaurant -----*/

R_Name varchar(50),

R_Address varchar (80),

Rate float,

GSTIN Varchar(30),

OPENING_TIME TIME,

CLOSING_TIME TIME,

phone bigint check(len(phone)=10)

)

/*----- Here inserting the value inside Restaurant----- */

insert into Restaurant values(101,'Sagar Gaire', '101, 7 Number Chouraha,
Shivaji Nagar, Bhopal, Madhya Pradesh 462016',4.2,'23AIHPG0018R2ZB','11:00','23:00', 07848011111)

---HERE CREATED A TABLE NAMED SUPPLIER TO STORE SUPPLIER DATA---

```
CREATE TABLE SUPPLIER(  
    SUPPLIER_ID INT NOT NULL PRIMARY KEY, /*----- creating SUPPLIER_ID as primary  
key for SUPPLIER -----*/  
    SUPPLIER_NAME VARCHAR(30),  
    SUPPLIER_ADDRESS VARCHAR(80),  
    SUPPLIER_PHONE bigINT CHECK(LEN(SUPPLIER_PHONE)=10),  
    EMAIL VARCHAR(40) ,  
    Restaurant__ID INT, /*----- foreign key Restaurant__id ----- References Restaurant  
-----*/
```

```
    FOREIGN KEY (RESTAURANT__ID) REFERENCES Restaurant(R_ID) /*----- Here  
Creating the Foreign key of Restaurant_id----- */ )
```

/*----- Here inserting the value inside Supplier----- */

```
insert into SUPPLIER values(301,'Rohit Singh','Goregow Bhaopal',8677930209,'rohitsingh24@gmail.com',101)  
insert into SUPPLIER values(302,'Aman Sahu','Nehru nager, Bhoapl',9644930209,'Amansah769@gmail.com',101)  
insert into SUPPLIER values(303,'Ashu Sharma','Rohit Nager, Bhopal',6263240209,'Ashu7689@gmail.com',101)  
insert into SUPPLIER values(304,'Moahit Mishra','Anand Nager Bhoapl',7828930209,'rohitsingh88@gmail.com',101)  
insert into SUPPLIER values(305,'jay Prakash','MP nager, Bhopal',9807930209,'prakash675@gmail.com',101)
```

---HERE CREATED A TABLE NAMED WAITER TO STORE WAITER DETAILS---

```
Create table waiter(  
    W_ID int not null primary key, /*----- creating W_ID as primary key for waiter ----  
---*/  
    W_NAME VARCHAR(40),  
    GENDER VARCHAR(20),  
    D_O_B DATE ,  
    W_ADDRESS VARCHAR(80),  
    SALARY INT ,  
    RESTAURANT_ID INT, /*----- foreign key Restaurant_id ----- References  
Restaurant -----*/
```

```
    FOREIGN KEY (RESTAURANT_ID) REFERENCES Restaurant(R_ID) /*----- Here we  
are Creating the Foregin key of Restaurant_id----- */ )
```

/*----- Here inserting the value inside Waiter----- */

```
insert into waiter values(201, 'Amit Kumar', 'Male', '1995-05-15', '123 Main Street, Delhi', 20000, 101)  
insert into waiter values(202, 'Sneha Sharma', 'Female', '1992-09-30', '456 Park Avenue, Mumbai', 18000, 101)
```

```

insert into waiter values(203, 'Rajesh Singh', 'Male', '1988-12-10', '789 Oak Road, Bangalore', 22000, 101)
insert into waiter values(204, 'Priya Patel', 'Female', '1995-02-22', '101 Elm Street, Chennai', 19000, 101)
insert into waiter values(205, 'Nikhil Verma', 'Male', '1991-07-05', '222 Maple Lane, Kolkata', 21000, 101)
insert into waiter values(206, 'Anjali Sharma', 'Female', '1993-11-12', '333 Pine Avenue, Hyderabad', 19500, 101)
insert into waiter values(207, 'Vikas Gupta', 'Male', '1994-08-20', '444 Cedar Road, Pune', 20500, 101)
insert into waiter values(208, 'Divya Khanna', 'Female', '1990-04-17', '555 Oak Street, Jaipur', 18500, 101)
insert into waiter values(209, 'Rahul Kapoor', 'Male', '1996-06-25', '666 Elm Lane, Ahmedabad', 19000, 101)
insert into waiter values(210, 'Sonia Singh', 'Female', '1992-03-08', '777 Maple Avenue, Chandigarh', 21500, 101)
insert into waiter values(211, 'Arun Sharma', 'Male', '1989-10-31', '888 Pine Road, Lucknow', 22500, 101)
insert into waiter values(212, 'Meena Rajput', 'Female', '1997-09-16', '999 Cedar Lane, Indore', 17500, 101)
insert into waiter values(213, 'Suresh Yadav', 'Male', '1993-12-01', '111 Oak Street, Kochi', 20500, 101)
insert into waiter values(214, 'Reena Nair', 'Female', '1994-02-14', '222 Elm Road, Surat', 18000, 101)
insert into waiter values(215, 'Sanjay Verma', 'Male', '1991-05-18', '333 Maple Lane, Bhopal', 19500, 101)
insert into waiter values(216, 'Pooja Gupta', 'Female', '1995-08-11', '444 Pine Avenue, Varanasi', 20000, 101)
insert into waiter values(217, 'Gaurav Kumar', 'Male', '1988-11-30', '555 Cedar Road, Coimbatore', 23000, 101)
insert into waiter values(218, 'Anita Singh', 'Female', '1992-07-09', '666 Oak Street, Visakhapatnam', 18500, 101)
insert into waiter values(219, 'Rajesh Yadav', 'Male', '1990-03-25', '777 Elm Lane, Nagpur', 21000, 101)
insert into waiter values(220, 'Preeti Sharma', 'Female', '1996-01-22', '888 Maple Avenue, Kanpur', 20000, 101)

```

----HERE CREATE A TABLE NAMED CUSTOMER FOR STORE CUSTOMER DATA----

```

CREATE TABLE CUSTOMER(
    C_ID INT NOT NULL PRIMARY KEY, /*----- creating C_ID as primary key for
CUSTOMER -----*/

    C_NAME VARCHAR(30),
    C_ADDRESS VARCHAR(80),
    C_DATE DATE,
    C_EMAIL VARCHAR(30),
    C_PHONE BIGINT CHECK(LEN(C_PHONE)=10),
    TABLE_ID INT, /*----- foreign key TABLE_ID ----- References SITING_TABLE ----
-----*/

    C_TIME TIME,
    REST_ID INT, /*----- foreign key REST_ID ----- References Restaurant -----
*/

    FOREIGN KEY (REST_ID) REFERENCES Restaurant(R_ID), /*----- Here we are
Creating the Foreign key of Restaurant_id----- */

    FOREIGN KEY (TABLE_ID) REFERENCES SITING_TABLE(T_ID) /*----- Here we are
Creating the Foreign key of Table_id----- */

```

)

alter table customer

drop column c_Date --- Dropping column date from customer

alter table customer

drop column c_time --- Dropping column TIME from customer

/*----- Here inserting the value inside CUSTOMER----- */

INSERT INTO CUSTOMER (C_ID, C_NAME, C_ADDRESS, C_EMAIL, C_PHONE, TABLE_ID, C_TIME, REST_ID)

VALUES

(701, 'Satish kumran', '123 Main Street, Bhopal', 'kumran76@gmail.com', 9876543210, 31, '19:30:00', 101),
(702, 'Akash Verma', '456 Park Avenue, Bhopal', 'akverma09@gmail.com', 8765432109, 32, '20:00:00', 101),
(703, 'Jagdeep singh', '789 Oak Road, Bhopal', 'Jagdeepsg3@gmail.com', 7654321098, 33, '19:45:00', 101),
(704, 'Sanu Bhardwaj', '101 Elm Street, Bhopal', 'BhardwajSanu9@gmail.com', 6543210987, 34, '21:00:00', 101),
(705, 'Ramesh Banarji', '222 Maple Lane, Bhopal', 'Ramesh623@gmail.com', 6232109876, 35, '19:15:00', 101),
(706, 'Harsh Davilas', '333 Pine Avenue, Bhopal', 'davilas34@gmail.com', 8321098765, 36, '20:30:00', 101),
(707, 'lalu sharma', '444 Cedar Road, Bhopal', 'sharmalalu88@gmail.com', 9210987654, 37, '19:45:00', 101),
(708, 'Abha Malviya', '555 Oak Street, Bhopal', 'abha8675@gmail.com', 7709876543, 38, '21:15:00', 101),
(709, 'Nupur Chourisiya', '666 Elm Lane, Bhopal', 'Chourisiyanupur09@gmail.com', 8598765432, 39, '20:30:00', 101),
(710, 'Isha Negi', '777 Maple Avenue, Bhopal', 'Isha67negi@gmail.com', 9876543210, 40, '21:30:00', 101),
(711, 'Jay Shree', '888 Pine Road, Lucknow', 'Jayshree7783@gmail.com', 8765432109, 31, '19:45:00', 101),
(712, 'Ankesh Malya', '999 Cedar Lane, Indore', 'Malyaankesh21@gmail.com', 7654321098, 32, '20:30:00', 101),
(713, 'Lokesh Patel', '111 Oak Street, Bhopal', 'Lokeshpt71@gmail.com', 6543210987, 33, '19:30:00', 101),
(714, 'Aman Khan', '222 Elm Road, Surat', 'Amankhan765@gmail.com', 9432109876, 34, '21:00:00', 101),
(715, 'Mason Gupta', '333 Maple Lane, Bhopal', 'masongupta875@gmail.com', 6654098765, 35, '19:00:00', 101),
(716, 'Harsh Singh', '444 Pine Avenue, Varanasi', 'Harshsinghbaghel66@gmail.com', 9955987654, 36, '20:15:00', 101),
(717, 'kamla Verma', '555 Cedar Road, Coimbatore', 'kamlavrma1107@gmail.com', 7654876543, 37, '19:30:00', 101),
(718, 'Saket Khan', '666 Oak Street, Visakhapatnam', 'Saketkh7an@gmail.com', 9876543209, 38, '21:45:00', 101),
(719, 'Prince Badhoriya', '777 Elm Lane, Nagpur', 'Badhoriyaprince00998@gmail.com', 9876543210, 39, '20:45:00', 101),
(720, 'Lily Sharma', '888 Maple Avenue, Kanpur', 'lily54@gmail.com', 8765432109, 40, '21:30:00', 101);

---AGAIN INSERT THE VALUES ---

INSERT INTO CUSTOMER (C_ID, C_NAME, C_ADDRESS, C_EMAIL, C_PHONE, TABLE_ID, C_TIME, REST_ID)

VALUES

(721, 'Rajesh Sharma', '123 ABC Street, Delhi', 'rajesh.sharma@example.com', 9876543211, 31, '19:30:00', 101),
(722, 'Shalini Gupta', '456 XYZ Avenue, Mumbai', 'shalini.gupta@example.com', 8765432110, 32, '20:00:00', 101),

(723, 'Amit Verma', '789 PQR Road, Kolkata', 'amit.verma@example.com', 7654321109, 33, '19:45:00', 101),
 (724, 'Meera Singh', '101 LMN Lane, Chennai', 'meera.singh@example.com', 6543211098, 34, '21:00:00', 101),
 (725, 'Rohit Kumar', '222 EFG Street, Bengaluru', 'rohit.kumar@example.com', 5432109876, 36, '19:15:00', 101),
 (726, 'Riya Patel', '333 GHI Avenue, Hyderabad', 'riya.patel@example.com', 4321098765, 35, '20:30:00', 101),
 (727, 'Sanjay Mishra', '444 UVW Road, Pune', 'sanjay.mishra@example.com', 3210987654, 37, '19:45:00', 101),
 (728, 'Deepa Chopra', '555 RST Lane, Ahmedabad', 'deepa.chopra@example.com', 2109876543, 39, '21:15:00', 101),
 (729, 'Nikhil Das', '666 XYZ Street, Jaipur', 'nikhil.das@example.com', 1098765432, 33, '20:30:00', 101),
 (730, 'Priya Choudhary', '777 JKL Avenue, Lucknow', 'priya.choudhary@example.com', 9876543211, 40, '21:30:00', 101),
 (731, 'Anshul Agarwal', '888 ABC Road, Delhi', 'anshul.agarwal@example.com', 8765432110, 37, '19:45:00', 101),
 (732, 'Alisha Jain', '999 XYZ Avenue, Mumbai', 'alisha.jain@example.com', 7654321109, 32, '20:30:00', 101),
 (733, 'Neha Malhotra', '111 PQR Lane, Kolkata', 'neha.malhotra@example.com', 6543211098, 33, '19:30:00', 101),
 (734, 'Kunal Rajput', '222 LMN Street, Chennai', 'kunal.rajput@example.com', 5432109876, 37, '21:00:00', 101),
 (735, 'Ritu Mehta', '333 EFG Avenue, Bengaluru', 'ritu.mehta@example.com', 4321098765, 39, '19:00:00', 101),
 (736, 'Aditya Shukla', '444 UVW Road, Hyderabad', 'aditya.shukla@example.com', 3210987654, 36, '20:15:00', 101),
 (737, 'Juhi Patel', '555 RST Lane, Pune', 'juhi.patel@example.com', 2109876543, 38, '19:30:00', 101),
 (738, 'Rahul Sharma', '666 XYZ Street, Ahmedabad', 'rahul.sharma@example.com', 1098765432, 31, '21:45:00', 101),
 (739, 'Sonam Das', '777 JKL Avenue, Jaipur', 'sonam.das@example.com', 9876543211, 33, '20:45:00', 101),
 (740, 'Alok Singh', '888 ABC Road, Lucknow', 'alok.singh@example.com', 8765432110, 40, '21:30:00', 101);

---HERE CREATE A TABLE NAMED SITING_TABLE FOR STORE TABLE DETAILS---

```
CREATE TABLE SITING_TABLE(
    T_ID INT NOT NULL PRIMARY KEY, /*----- creating T_ID as primary key for
SITING_TABLE -----*/
    CAPACITY INT,
    T_STATUS VARCHAR(20),
    RES_ID INT, /*----- foreign key RES_ID ----- References Restaurant -----
-*/
    FOREIGN KEY (RES_ID) REFERENCES Restaurant(R_ID) /*----- Here we are
Creating the Foreign key of Restaurant_id----- */ )
/*----- Here inserting the value inside Siting Table----- */
```

```
insert into SITING_TABLE values(31, 4, 'Occupied', 101)
insert into SITING_TABLE values(32, 2, 'Vacant', 101)
insert into SITING_TABLE values(33, 6, 'Occupied', 101)
insert into SITING_TABLE values(34, 8, 'Reserved', 101)
insert into SITING_TABLE values(35, 4, 'Vacant', 101)
```

```

insert into SITING_TABLE values(36, 2, 'Vacant', 101)
insert into SITING_TABLE values(37, 4, 'Occupied', 101)
insert into SITING_TABLE values(38, 6, 'Reserved', 101)
insert into SITING_TABLE values(39, 2, 'Vacant', 101)
insert into SITING_TABLE values(40, 4, 'Vacant', 101)

```

---HERE I AM CREATE A TABLE NAME MENU FOR STORE THE DISH DETAILS---

```

CREATE TABLE MENU(

```

```

    DISH_ID INT PRIMARY KEY, /*----- creating DISH_ID as primary key for MENU -----*/

```

```

    DISH_NAME VARCHAR(30),

```

```

    DISH_TYPE VARCHAR(30),

```

```

    PRICE INT ,

```

```

    RESTR_ID INT, /*----- foreign key RESTR_ID ----- References Restaurant -----*/

```

```

    FOREIGN KEY (RESTR_ID) REFERENCES Restaurant(R_ID) /*----- Here Creating the Foreign
key of Restaurant_id----- */ )

```

/*----- Here inserting the value inside MENU ----- */

```

INSERT INTO MENU (DISH_ID, DISH_NAME, DISH_TYPE, PRICE, RESTR_ID)

```

```

VALUES

```

```

(1, 'Butter Chicken', 'Non-Vegetarian', 300, 101),
(2, 'Palak Paneer', 'Vegetarian', 250, 101),
(3, 'Masala Dosa', 'Vegetarian', 150, 101),
(4, 'Biryani', 'Non-Vegetarian', 350, 101),
(5, 'Chole Bhature', 'Vegetarian', 180, 101),
(6, 'Idli Sambar', 'Vegetarian', 120, 101),
(7, 'Rogan Josh', 'Non-Vegetarian', 320, 101),
(8, 'Aloo Gobi', 'Vegetarian', 200, 101),
(9, 'Pani Puri', 'Vegetarian', 100, 101),
(10, 'Fish Curry', 'Non-Vegetarian', 280, 101),
(11, 'Chicken Tikka', 'Non-Vegetarian', 280, 101),
(12, 'Paneer Tikka', 'Vegetarian', 220, 101),
(13, 'Rasgulla', 'Vegetarian', 80, 101),
(14, 'Mutton Biryani', 'Non-Vegetarian', 400, 101),
(15, 'Dal Makhani', 'Vegetarian', 180, 101),
(16, 'Veg Pulao', 'Vegetarian', 200, 101),
(17, 'Tandoori Chicken', 'Non-Vegetarian', 320, 101),

```



```
(18, 'Gulab Jamun', 'Vegetarian', 90, 101),
(19, 'Chole Kulche', 'Vegetarian', 150, 101),
(20, 'Fish Fry', 'Non-Vegetarian', 240, 101),
(21, 'Mango Lassi', 'Beverage', 120, 101),
(22, 'Masala Chai', 'Beverage', 90, 101),
(23, 'Cold Coffee', 'Beverage', 150, 101),
(24, 'Fresh Lime Soda', 'Beverage', 80, 101),
(25, 'Buttermilk', 'Beverage', 70, 101);
```

---HERE I AM CREATING A TABLE CUSTOMER_TIME FOR STORING CUSTOMER TIME---

```
CREATE TABLE CUSTOMER_TIME(
    CUSTOMER_ID INT, /*----- foreign key Customer_id ----- References Customer -----
    -----*/
    CHECKIN_TIME TIME,
    CHECKOUT_TIME TIME,
    FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMER(C_ID) /*----- Here
Creating the Foreign key of Customer_id----- */
)
```

```
ALTER TABLE CUSTOMER_TIME
ADD DATE DATE,
DAY VARCHAR(20); --- ADDING DATE AND DAY USING ALTER IN CUSTOMER_TIME
/*----- Here inserting the value inside CUSTOMER_TIME----- */
```

```
INSERT INTO CUSTOMER_TIME (CUSTOMER_ID, CHECKIN_TIME, CHECKOUT_TIME)
VALUES
(701, '19:30:00', '21:45:00'),
(702, '20:00:00', '22:30:00'),
(703, '19:45:00', '21:15:00'),
(704, '21:00:00', '23:00:00'),
(705, '19:15:00', '21:00:00'),
(706, '20:30:00', '22:00:00'),
(707, '19:45:00', '21:35:00'),
(708, '21:15:00', '23:10:00'),
(709, '20:30:00', '22:20:00'),
(710, '21:30:00', '23:30:00'),
(711, '19:45:00', '21:50:00'),
(712, '20:30:00', '22:15:00'),
```

```
(713, '19:30:00', '21:25:00'),  
(714, '21:00:00', '23:15:00'),  
(715, '19:00:00', '20:45:00'),  
(716, '20:15:00', '22:00:00'),  
(717, '19:30:00', '21:35:00'),  
(718, '21:45:00', '23:30:00'),  
(719, '20:45:00', '22:30:00'),  
(720, '21:30:00', '23:00:00');
```

---AGAIN INSERTING THE VALUES---

```
INSERT INTO CUSTOMER_TIME (CHECKIN_TIME, CHECKOUT_TIME, DATE, DAY, CUSTOMER_ID)  
VALUES
```

```
('09:30:00', '11:45:00', '2023-08-05', 'Friday', 721),  
( '12:15:00', '14:30:00', '2023-08-06', 'Saturday', 722),  
( '17:00:00', '19:30:00', '2023-08-07', 'Sunday', 723),  
( '15:45:00', '18:00:00', '2023-08-08', 'Monday', 724),  
( '20:15:00', '21:45:00', '2023-08-09', 'Tuesday', 725),  
( '11:30:00', '13:00:00', '2023-08-10', 'Wednesday', 726),  
( '14:45:00', '16:30:00', '2023-08-11', 'Thursday', 727),  
( '19:00:00', '20:45:00', '2023-08-12', 'Friday', 728),  
( '09:15:00', '10:30:00', '2023-08-13', 'Saturday', 729),  
( '12:30:00', '14:00:00', '2023-08-14', 'Sunday', 730),  
( '17:45:00', '19:15:00', '2023-08-15', 'Monday', 731),  
( '14:00:00', '15:45:00', '2023-08-16', 'Tuesday', 732),  
( '16:30:00', '18:00:00', '2023-08-17', 'Wednesday', 733),  
( '20:00:00', '22:00:00', '2023-08-18', 'Thursday', 734),  
( '11:15:00', '13:30:00', '2023-08-19', 'Friday', 735),  
( '13:45:00', '15:15:00', '2023-08-20', 'Saturday', 736),  
( '18:30:00', '20:00:00', '2023-08-21', 'Sunday', 737),  
( '15:00:00', '16:45:00', '2023-08-22', 'Monday', 738),  
( '17:30:00', '19:00:00', '2023-08-23', 'Tuesday', 739),  
( '20:30:00', '22:15:00', '2023-08-24', 'Wednesday', 740);
```

---UPDATING THE TIME USING UPDATE COMMAND---

```
update CUSTOMER_TIME set DATE ='2023-08-03'  
where CUSTOMER_ID between 701 and 703
```

```
update CUSTOMER_TIME set DATE ='2023-08-04'
      where CUSTOMER_ID between 704 and 707
```

```
update CUSTOMER_TIME set DAY ='Friday'
where CUSTOMER_ID between 704 and 707
```

```
update CUSTOMER_TIME set DATE ='2023-08-05'
      where CUSTOMER_ID between 708 and 711
```

```
      update CUSTOMER_TIME set DAY ='Saturday'
where CUSTOMER_ID between 708 and 711
```

```
update CUSTOMER_TIME set DATE ='2023-08-05'
      where CUSTOMER_ID between 712 and 715
```

```
update CUSTOMER_TIME set DAY ='Sunday'
where CUSTOMER_ID between 712 and 715
```

```
update CUSTOMER_TIME set DATE ='2023-08-05'
      where CUSTOMER_ID between 716 and 720
```

```
update CUSTOMER_TIME set DAY ='Monday'
where CUSTOMER_ID between 716 and 720
```

---HERE I AM CREATE A TABLE NAMED ORDER TO STORE OREDR DETAILS---

```
CREATE TABLE ORDERS(
      ORDER_ID INT NOT NULL PRIMARY KEY, /*----- creating ORDER_ID as primary key for
ORDERS -----*/
      DISH_ID INT , /*----- foreign key DISH_id ----- References Menu -----*/
      COOKING_TIME VARCHAR(20),
      PRICE INT ,
      CUS_ID INT, /*----- foreign key Customer_id ----- References Customer -----*/
      WAITER_ID INT, /*----- foreign key Waiter_id ----- References Waiter -----*/
      FOREIGN KEY (CUS_ID ) REFERENCES CUSTOMER(C_ID), /*----- Here we are Creating the
Foreign key of Customer_id----- */
```

```
FOREIGN KEY (WAITER_ID ) REFERENCES waiter(W_ID), /*----- Here we are Creating the  
Foreign key of Waiter_id----- */
```

```
FOREIGN KEY (DISH_ID) REFERENCES MENU(DISH_ID) /*----- Here we are Creating the  
Foregin key of Dish_id----- */ )
```

```
/*----- Here I am inserting the value inside Orders ----- */
```

```
INSERT INTO ORDERS (ORDER_ID, DISH_ID, COOKING_TIME, PRICE, CUS_ID, WAITER_ID)
```

```
VALUES
```

```
(401, 1, '20 minutes', 300, 701, 201),  
(402, 5, '15 minutes', 180, 702, 204),  
(403, 3, '10 minutes', 150, 703, 203),  
(404, 8, '12 minutes', 200, 704, 204),  
(405, 4, '25 minutes', 350, 705, 206),  
(406, 6, '18 minutes', 120, 706, 201),  
(407, 2, '14 minutes', 250, 707, 205),  
(408, 10, '30 minutes', 280, 708, 203),  
(409, 7, '22 minutes', 320, 709, 207),  
(410, 9, '20 minutes', 100, 710, 209),  
(411, 1, '19 minutes', 280, 711, 208),  
(412, 5, '16 minutes', 190, 712, 202),  
(413, 3, '11 minutes', 160, 713, 206),  
(414, 8, '14 minutes', 210, 714, 204),  
(415, 4, '24 minutes', 340, 715, 205),  
(416, 6, '17 minutes', 110, 716, 201),  
(417, 2, '13 minutes', 240, 717, 203),  
(418, 10, '29 minutes', 270, 718, 209),  
(419, 7, '21 minutes', 310, 719, 207),  
(420, 9, '18 minutes', 90, 720, 205),  
(421, 1, '20 minutes', 300, 701, 201),  
(422, 5, '15 minutes', 180, 701, 202),  
(423, 3, '10 minutes', 150, 701, 207),  
(424, 8, '12 minutes', 200, 702, 204),  
(425, 4, '25 minutes', 350, 703, 207),  
(426, 6, '18 minutes', 120, 704, 201),
```

(427, 2, '14 minutes', 250, 705, 206),
(428, 10, '30 minutes', 280, 706, 208),
(429, 7, '22 minutes', 320, 707, 204),
(430, 9, '20 minutes', 100, 708, 209),
(431, 1, '19 minutes', 280, 709, 201),
(432, 5, '16 minutes', 190, 710, 202),
(433, 3, '11 minutes', 160, 711, 205),
(434, 8, '14 minutes', 210, 712, 204),
(435, 4, '24 minutes', 340, 713, 209),
(436, 6, '17 minutes', 110, 714, 208),
(437, 2, '13 minutes', 240, 715, 207),
(438, 10, '29 minutes', 270, 716, 203),
(439, 7, '21 minutes', 310, 717, 204),
(440, 9, '18 minutes', 90, 718, 205);

---AGAIN INSERTING THE VALUE ---

INSERT INTO ORDERS (ORDER_ID, DISH_ID, COOKING_TIME, PRICE, CUS_ID, WAITER_ID)

VALUES

(441, 1, '15 minutes', 300, 721, 201),
(442, 2, '18 minutes', 250, 722, 202),
(443, 3, '20 minutes', 150, 723, 203),
(444, 4, '22 minutes', 350, 724, 204),
(445, 5, '25 minutes', 180, 725, 205),
(446, 6, '30 minutes', 120, 726, 206),
(447, 7, '24 minutes', 320, 727, 207),
(448, 8, '28 minutes', 200, 728, 208),
(449, 9, '26 minutes', 100, 729, 209),
(450, 10, '19 minutes', 280, 730, 210),
(451, 11, '22 minutes', 280, 731, 201),
(452, 12, '18 minutes', 220, 732, 202),
(453, 13, '15 minutes', 80, 733, 203),
(454, 14, '30 minutes', 400, 734, 204),
(455, 15, '28 minutes', 180, 735, 205),
(456, 16, '25 minutes', 200, 736, 206),
(457, 17, '20 minutes', 320, 737, 207),
(458, 18, '22 minutes', 90, 738, 208),

(459, 19, '19 minutes', 150, 739, 209),

(460, 20, '28 minutes', 240, 740, 210);

select * from ORDERS

---HERE I AM CREATE A TABLE NAMED PAYMENT FOR STORING THE DATA OF PAYMENT---

CREATE TABLE PAYMENT(

PAYMENT_ID INT PRIMARY KEY, /*----- creating PAYMENT_ID as primary key for
PAYMENT -----*/

AMOUNT INT ,

PAYMENT_DATE DATE,

PAYMENT_METHOD VARCHAR(30),

CUSTOMER__ID INT, /*----- foreign key Customer__id ----- References Customer
-----*/

FOREIGN KEY (CUSTOMER__ID) REFERENCES CUSTOMER(C_ID), /*----- Here we are
Creating the Foreign key of Customer_id----- */)

/*----- Here we are inserting the value inside PAYMENT----- */

INSERT INTO PAYMENT (PAYMENT_ID, AMOUNT, PAYMENT_DATE, PAYMENT_METHOD, CUSTOMER__ID)

VALUES

(111, 930, '2023-08-03', 'Credit Card', 701),

(112, 380, '2023-08-03', 'Cash', 702),

(113, 500, '2023-08-03', 'Debit Card', 703),

(114, 320, '2023-08-03', 'Cash', 704),

(115, 600, '2023-08-03', 'Credit Card', 705),

(116, 400, '2023-08-03', 'Cash', 706),

(117, 570, '2023-08-03', 'Debit Card', 707),

(118, 380, '2023-08-03', 'Cash', 708),

(119, 600, '2023-08-03', 'Credit Card', 709),

(120, 290, '2023-08-03', 'Cash', 710),

(121, 440, '2023-08-03', 'Credit Card', 711),

(122, 400, '2023-08-03', 'Cash', 712),

(123, 500, '2023-08-03', 'Debit Card', 713),

(124, 210, '2023-08-03', 'Cash', 714),

(125, 580, '2023-08-03', 'Credit Card', 715),

(126, 380, '2023-08-03', 'Cash', 716),

(127, 550, '2023-08-03', 'Debit Card', 717),

(128, 360, '2023-08-03', 'Cash', 718),

(129, 310, '2023-08-03', 'Credit Card', 719),

(130, 90, '2023-08-03', 'Cash', 720);

----Here again inserting the Values ----

INSERT INTO PAYMENT (PAYMENT_ID, PAYMENT_METHOD, PAYMENT_DATE, time, AMOUNT, CUSTOMER_ID)

VALUES

(131, 'Credit Card', '2023-08-05', '09:30:00', 300, 721),

(132, 'Cash', '2023-08-06', '12:15:00', 250, 722),

(133, 'Debit Card', '2023-08-07', '17:00:00', 150, 723),

(134, 'Cash', '2023-08-08', '15:45:00', 350, 724),

(135, 'Credit Card', '2023-08-09', '20:15:00', 180, 725),

(136, 'Cash', '2023-08-10', '11:30:00', 120, 726),

(137, 'Debit Card', '2023-08-11', '14:45:00', 320, 727),

(138, 'Credit Card', '2023-08-12', '19:00:00', 200, 728),

(139, 'Cash', '2023-08-13', '09:15:00', 100, 729),

(140, 'Credit Card', '2023-08-14', '12:30:00', 280, 730),

(141, 'Debit Card', '2023-08-15', '17:45:00', 280, 731),

(142, 'Cash', '2023-08-16', '14:00:00', 220, 732),

(143, 'Credit Card', '2023-08-17', '16:30:00', 80, 733),

(144, 'Debit Card', '2023-08-18', '20:00:00', 400, 734),

(145, 'Cash', '2023-08-19', '11:15:00', 180, 735),

(146, 'Credit Card', '2023-08-20', '13:45:00', 200, 736),

(147, 'Cash', '2023-08-21', '18:30:00', 320, 737),

(148, 'Debit Card', '2023-08-22', '15:00:00', 90, 738),

(149, 'Credit Card', '2023-08-23', '17:30:00', 150, 739),

(150, 'Cash', '2023-08-24', '20:30:00', 240, 740);

---- Adding column time using Alter

Alter table payment

add Time Time

---- Updating the time----

update payment set time = '20:35:00' ---updating time --

where CUSTOMER_ID = 701

update payment set time = '22:38:00' ---updating time --

where CUSTOMER_ID = 702

update payment set time = '21:35:00' ---updating time --

where CUSTOMER_ID = 703

```

update payment set time = '23:00:00' ---updating time --
    where CUSTOMER___ID =704
update payment set time = '20:55:00' ---updating time --
    where CUSTOMER___ID =705
update payment set time = '21:40:00' ---updating time --
    where CUSTOMER___ID =706
update payment set time = '20:00:00' ---updating time --
    where CUSTOMER___ID =707
update payment set time = '21:25:00' ---updating time --
    where CUSTOMER___ID =708
update payment set time = '22:15:00' ---updating time --
    where CUSTOMER___ID =709
update payment set time = '23:25:00' ---updating time --
    where CUSTOMER___ID =710
update payment set time = '21:49:00' ---updating time --
    where CUSTOMER___ID =711
update payment set time = '22:05:00' ---updating time --
    where CUSTOMER___ID =712
update payment set time = '21:07:00' ---updating time --
    where CUSTOMER___ID =713
update payment set time = '23:10:00' ---updating time --
    where CUSTOMER___ID =714
update payment set time = '20:30:00' ---updating time --
    where CUSTOMER___ID =715
update payment set time = '21:57:00' ---updating time --
    where CUSTOMER___ID =716
update payment set time = '20:48:00' ---updating time --
    where CUSTOMER___ID =717
update payment set time = '23:19:00' ---updating time --
    where CUSTOMER___ID =718
update payment set time = '22:09:00' ---updating time --
    where CUSTOMER___ID =719
update payment set time = '22:53:00' ---updating time --
    where CUSTOMER___ID =720

```

---- Here I am creating a table Feedback for Storing Customer Feedback----


```

CREATE TABLE FEEDBACK(
    CUSTOMER__ID INT , /*----- foreign key Customer__id ----- References Customer -----
-----*/

    F_SERVICE VARCHAR(30),

    F_SUGGESTION VARCHAR(80),

    F_CLEANNES VARCHAR(30),

    FOOD_QUALITY VARCHAR(30),

    FOREIGN KEY (CUSTOMER__ID ) REFERENCES CUSTOMER(C_ID), /*----- Here we are
Creating the Foreign key of Customer_id----- */

)

```

```

/*----- Here I am inserting the value inside Feedback----- */

```

```

INSERT INTO FEEDBACK (CUSTOMER__ID, F_SERVICE, F_SUGGESTION, F_CLEANNES, FOOD_QUALITY)
VALUES

(701, 'Excellent', 'The service was quick and attentive.', 'Very clean and well-maintained.', 'Delicious food, loved it!'),

(702, 'Good', 'Service could be a bit faster.', 'Clean, but can be improved.', 'Tasty food overall.'),

(703, 'Excellent', 'No suggestions, everything was perfect.', 'Clean and well-organized.', 'Amazing food, highly
recommended!'),

(704, 'Average', 'Service needs improvement.', 'Fairly clean, but could be better.', 'Decent food, could be better.'),

(705, 'Good', 'Polite and friendly service.', 'Clean and tidy.', 'Good taste, enjoyed the meal.'),

(706, 'Excellent', 'Great service, very accommodating.', 'Spotless and hygienic.', 'Delicious and well-prepared.'),

(707, 'Average', 'Service was okay, but some delays.', 'Clean but some areas could be improved.', 'Decent food, not
outstanding.'),

(708, 'Good', 'Service could be faster.', 'Generally clean and presentable.', 'Enjoyed the meal.'),

(709, 'Excellent', 'Attentive and friendly staff.', 'Clean and well-maintained.', 'Delicious food, will visit again!'),

(710, 'Good', 'Service was good overall.', 'Cleanliness could be improved.', 'Tasty food, satisfied with the experience.'),

(711, 'Good', 'Service was good overall.', 'Clean and well-maintained.', 'Tasty food, satisfied with the experience.'),

(712, 'Average', 'Service was a bit slow.', 'Cleanliness could be better.', 'Food was okay, nothing exceptional.'),

(713, 'Good', 'Polite and helpful service.', 'Clean and well-organized.', 'Good food, enjoyed the flavors.'),

(714, 'Excellent', 'Excellent service and attention.', 'Very clean and hygienic.', 'Delicious food, loved the variety.'),

(715, 'Average', 'Service could be more attentive.', 'Clean, but some areas need improvement.', 'Average food, nothing
extraordinary.'),

(716, 'Good', 'Friendly and prompt service.', 'Clean and tidy.', 'Good taste, enjoyed the meal.'),

(717, 'Good', 'Service was good overall.', 'Clean and well-maintained.', 'Tasty food, satisfied with the experience.'),

(718, 'Average', 'Service needs improvement.', 'Fairly clean, but could be better.', 'Decent food, could be better.'),

(719, 'Excellent', 'Attentive and friendly staff.', 'Clean and well-maintained.', 'Delicious food, will visit again!'),

(720, 'Good', 'Service was good overall.', 'Cleanliness could be improved.', 'Tasty food, satisfied with the experience.');
```

--- Again Inserting some feedbacks ---

```
INSERT INTO FEEDBACK (CUSTOMER__ID, F_SERVICE, F_SUGGESTION, F_CLEANNES, FOOD_QUALITY)
VALUES
(721, 'Good', 'The service was prompt and courteous.', 'Clean and well-maintained.', 'Delicious food, loved it!'),
(722, 'Excellent', 'The staff was very attentive and helpful.', 'Spotlessly clean.', 'Amazing food, will come again!'),
(723, 'Satisfactory', 'The service could be improved.', 'Average cleanliness.', 'Food was okay, nothing special.'),
(724, 'Poor', 'The service needs improvement.', 'Not very clean.', 'The food was just average.'),
(725, 'Good', 'Service was decent.', 'Clean but could be better.', 'Tasty food, good experience.'),
(726, 'Excellent', 'The staff was very friendly.', 'Clean and well-organized.', 'Delicious and well-presented dishes.'),
(727, 'Satisfactory', 'The service was average.', 'Could be cleaner.', 'Food was decent, nothing extraordinary.'),
(728, 'Good', 'The staff was attentive.', 'Fairly clean.', 'Enjoyed the food, it was flavorful.'),
(729, 'Poor', 'The service needs improvement.', 'Not very clean.', 'The food was just average.'),
(730, 'Good', 'Service was efficient.', 'Clean and tidy.', 'Food was good, overall satisfied.'),
(731, 'Excellent', 'The staff was very courteous.', 'Clean and hygienic.', 'Delicious food, loved the flavors!'),
(732, 'Satisfactory', 'The service was okay.', 'Could be cleaner.', 'Food was decent, but could be better.'),
(733, 'Poor', 'Service was slow and inattentive.', 'Not clean at all.', 'Food was disappointing, not come again.'),
(734, 'Good', 'Service was decent.', 'Cleanliness was satisfactory.', 'The food was good, satisfied overall.'),
(735, 'Excellent', 'The staff was very friendly and helpful.', 'Clean and well-maintained.', 'Delicious and well-prepared food.'),
(736, 'Satisfactory', 'The service could be better.', 'Clean but needs improvement.', 'Food was okay, nothing special.'),
(737, 'Good', 'The staff was attentive.', 'Clean and tidy.', 'Enjoyed the food, it was flavorful.'),
(738, 'Poor', 'Service was slow and unprofessional.', 'Not very clean.', 'Food was disappointing, recommend.'),
(739, 'Good', 'Service was efficient.', 'Clean and hygienic.', 'Food was good, overall satisfied.'),
(740, 'Excellent', 'The staff was very courteous.', 'Clean and well-organized.', 'Delicious and well-presented dishes.');
```

QUERIES

Q1.How many times a customer can visited and what he orders ?

```
CREATE PROCEDURE SelectAllCustomers
AS
begin
SELECT CUSTOMER.C_ID, CUSTOMER.C_NAME, Date, DISH_NAME FROM CUSTOMER inner JOIN ORDERS
O ON CUSTOMER.C_ID = O.CUS_ID inner JOIN MENU M ON O.DISH_ID = M.DISH_ID
inner join CUSTOMER_TIME on CUSTOMER.C_ID =CUSTOMER_TIME.CUSTOMER_ID where date
between '2023-08-01' and '2023-08-31' group by CUSTOMER.C_ID, CUSTOMER.C_NAME, Date,
DISH_NAME order by COUNT(cus_id) desc
end;

EXEC SelectAllCustomers
```

OUTPUT

The screenshot shows a SQL Developer window with a query titled '17072023 query.s..._SAYAM KUAMR (70)'. The query is a stored procedure named 'SelectAllCustomers' that joins 'CUSTOMER', 'ORDERS', and 'MENU' tables. The results pane shows a table with 59 rows and 5 columns: C_ID, C_NAME, Date, and DISH_NAME. The data is sorted by C_ID in descending order.

C_ID	C_NAME	Date	DISH_NAME
701	Salish kumran	2023-08-03	Butter Chicken
701	Salish kumran	2023-08-03	Masala Dosa
702	Akash Verma	2023-08-03	Aloo Gobi
702	Akash Verma	2023-08-03	Chole Bhature
703	Jagdeep singh	2023-08-03	Biryani
703	Jagdeep singh	2023-08-03	Masala Dosa
704	Sanu Bhardwaj	2023-08-04	Aloo Gobi
704	Sanu Bhardwaj	2023-08-04	Idli Sambar
705	Ramesh Banarji	2023-08-04	Biryani
705	Ramesh Banarji	2023-08-04	Chole Bhature
705	Ramesh Banarji	2023-08-04	Palak Paneer
706	Harsh Davilas	2023-08-04	Fish Curry
706	Harsh Davilas	2023-08-04	Idli Sambar
707	Ishu sharma	2023-08-04	Palak Paneer
707	Ishu sharma	2023-08-04	Rogan Josh
708	Abha Malviya	2023-08-05	Fish Curry
708	Abha Malviya	2023-08-05	Pani Puri
709	Nupur Chouris...	2023-08-05	Butter Chicken
709	Nupur Chouris...	2023-08-05	Rogan Josh
710	Isha Negi	2023-08-05	Chole Bhature
710	Isha Negi	2023-08-05	Pani Puri
711	Jay Shree	2023-08-05	Butter Chicken
711	Jay Shree	2023-08-05	Masala Dosa
712	Ankesh Malviya	2023-08-05	Aloo Gobi
712	Ankesh Malviya	2023-08-05	Chole Bhature
713	Lokesh Patel	2023-08-05	Biryani
713	Lokesh Patel	2023-08-05	Masala Dosa
714	Aman Khan	2023-08-05	Aloo Gobi
714	Aman Khan	2023-08-05	Idli Sambar

Q2. How much revenue generate from every customers ?

CREATE PROCEDURE GetCustomerRevenue

AS

BEGIN

SELECT C_NAME, SUM(price) AS total_revenue FROM Orders inner join CUSTOMER on C_ID = CUS_ID group by C_NAME order by total_revenue desc

END;

EXEC GetCustomerRevenue

OUTPUT

The screenshot shows the SQL Developer window with the 'GetCustomerRevenue' procedure executed. The results pane displays a table with 40 rows and 2 columns: C_NAME and total_revenue. The results are sorted in descending order of total_revenue.

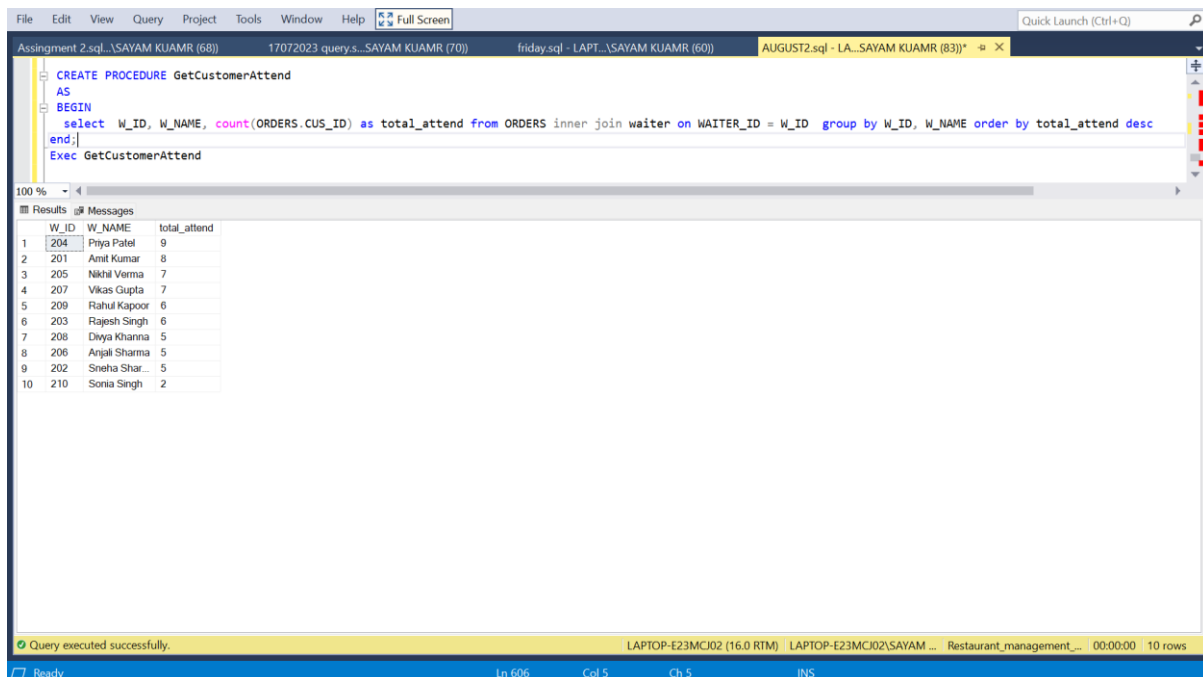
C_NAME	total_revenue
Ramesh Banarji	780
Salish kumran	750
Nupur Chouris...	650
Mason Gupta	580
Ishu sharma	570
Akash Verma	550
Jagdeep singh	500
Lokesh Patel	500
Jay Shree	440
Harsh Davilas	400
Kunal Ragut	400
Ankesh Malviya	400
Abha Malviya	380
Akash Verma	380
Harsh Singh	380
Saket Khan	360
Meera Singh	350
Sangay Mishra	320
Sanu Bhardwaj	320
Juhi Patel	320
Aman Khan	320
Prince Dadhony	310
Rajesh Sharma	300
Isha Negi	290
Anshul Agarwal	280
Priya Choudhary	280
Shalini Gupta	250
Alok Singh	240
Aishu Jain	220

Q3.How much customer served by a waiter ?

```
CREATE PROCEDURE GetCustomerAttend
AS
BEGIN
    select W_ID, W_NAME, count(ORDERS.CUS_ID) as total_attend from ORDERS inner join
    waiter on WAITER_ID = W_ID group by W_ID, W_NAME order by total_attend desc
end;
```

Exec GetCustomerAttend

OUTPUT



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the SQL code for the 'GetCustomerAttend' stored procedure. The bottom pane shows the results of the query, which is a table with three columns: W_ID, W_NAME, and total_attend. The results are ordered by total_attend in descending order. The status bar at the bottom indicates that the query was executed successfully.

	W_ID	W_NAME	total_attend
1	204	Priya Patel	9
2	201	Armit Kumar	8
3	205	Nikhil Verma	7
4	207	Vikas Gupta	7
5	209	Rahul Kapoor	6
6	203	Rajesh Singh	6
7	208	Divya Khanna	5
8	206	Anjali Sharma	5
9	202	Sneha Shar...	5
10	210	Sonia Singh	2

THANK YOU