Create database Zomato;

drop table if exists Zomato.customer;

create table customer(

customer\_id int(4) not null,

customer\_name varchar(20) not null,

contact\_number varchar(11) not null,

address varchar(30) not null,

constraint pk\_customer primary key (customer\_id)

);

insert into customer(customer\_id,customer\_name,contact\_number,address) VALUES

(1001,'vaibhav Palkar','1234567801','01 SND, aundh, pune'),

(1002,'pratik kijbil','1234567802','02 WBG, kothrud, pune'),

(1003,'sagar amritkar','1234567803','03 QWE, wakad, pune'),

(1004,'mitali basargi','1234567804','04 RTY, sanghavi, pune'),

(1005,'jagruti bhoite','1234567805','05 SGV, sanghavi, pune'),

(1006,'sandhya shirale','1234567806','06 ASD, ravet, pune'),

(1007,'uday patil','1234567807','07 BLA, baner, pune'),

(1008,'punam kale','1234567808','08 SNR, warje, pune'),

(1009,'vighnesh gopinath','1234567809','09 ABP, ravet, pune'),

(1010,'nilesh supekar','1234567810','13 MGV, moshi, pune'),

(1011,'Murtaza khn','1234567811','17 ABP, vadgaon, pune'),

(1012,'gopal kasat','1234567812','19 ABP, pashan, pune'),

(1013,'yogesh jadhav','1234567813','20 ABP, pimpri, pune'),

(1014,'apeksha jadhav','1234567814','21 ABP, sinhagad, pune'),

(1015,'harshad kunjir','1234567815','22 DVK, Uruli, pune'),

(1016,'manash konwar','1234567816','57 SND, kasarwadi, pune'),

(1017,'someshwar kale','1234567817','11 MKN, mukund nagar, pune');

select \* from customer;

drop table if exists restaurant;

create table restaurant(

restaurant\_id int(3) not null,

restaurant\_name varchar(20) not null,

rlocation varchar(20) not null,

rrating decimal(2,1) not null,

constraint pk\_customer primary key (restaurant\_id)

);

insert into restaurant(restaurant\_id,restaurant\_name,rlocation,rrating) VALUES

(101, 'hydrabadi spice', 'aundh', '4.5'),

(102, 'hotel green park', 'baner', '4.1'),

(103, 'saffron', 'pashan', '3.9'),

(104, 'thomson restro', 'sanghavi', '3.6'),

(105, 'laa unico', 'mukund nagar', '4.3'),

(106, 'blue water', 'ravet', '4.2'),

(107, 'jalsaa restaurant', 'wakad', '4.3');

select \* from restaurant;

create table zomato\_employee(

employee\_id int(6) not null,

employee\_name varchar(20) not null,

emp\_contact\_number varchar(11) not null,

employee\_avg\_rating decimal(2,1) not null,

constraint pk\_zomato\_employee primary key (employee\_id)

);

insert into zomato\_employee(employee\_id, employee\_name, emp\_contact\_number, employee\_avg\_rating) values

(210111, 'sudesh s', '123456911', '3.5'),

(210112, 'prashant kedare', '123456912', '3.9'),

(210113, 'jignesh satam', '123456913', '4.5'),

(210114, 'lajat sharma', '123456914', '4.2'),

(210115, 'prem chopda', '123456915', '3.2'),

(210116, 'ghanashyam seth', '123456916', '3.7'),

(210117, 'sadanad rathe', '123456917', '4.4'),

(210118, 'satish p', '123456918', '4.6'),

(210119, 'mansigh tomar', '123456919', '3.1'),

(210120, 'prasad kechad', '123456920', '3.6'),

(210121, 'sami j', '123456921', '4.3'),

(210122, 'praful patel', '123456922', '3.8'),

(210123, 'narayan kumar', '123456923', '4.7'),

(210124, 'ahmed Z', '123456924', '3.5'),

(210125, 'jateen kumar', '123456925', '4.1');

select \* from zomato\_employee;

create table foods(

food\_id int(7) not null,

food\_name varchar(20) not null,

price\_per\_unit decimal(5,2) not null,

constraint pk\_foods primary key (food\_id)

);

insert into foods(food\_id, food\_name, price\_per\_unit) values

(9999411, 'allo paratha', '80.00'),

(9999412, 'veg meal', '120.00'),

(9999413, 'chapati', '20.00'),

(9999414, 'veg thali', '220.00'),

(9999415, 'chicken thali', '150.00'),

(9999416, 'veg biryani', '280.00'),

(9999417, 'mix veg', '180.00'),

(9999418, 'veg pulav', '190.00'),

(9999419, 'mix paratha', '180.00'),

(9999420, 'cold drink', '35.00'),

(9999421, 'paneer roll', '180.00'),

(9999422, 'veg roll', '180.00');

select \* from foods;

drop table if exists order\_detail;

create table order\_detail(

order\_id int(4) not null primary key,

customer\_id int(4) not null,

restaurant\_id int(3) not null,

employee\_id int(6) not null,

order\_status varchar(10),

order\_time timestamp,

delivered\_time timestamp,

constraint fk\_customer foreign key (customer\_id) references customer(customer\_id),

constraint fk\_restaurant foreign key (restaurant\_id) references restaurant(restaurant\_id),

constraint fk\_zomato\_employee foreign key (employee\_id) references zomato\_employee(employee\_id)

);

-- constraint fk\_restaurant foreign key (restaurant\_id) references restaurant(restaurant\_id)

-- constraint fk\_zomato\_employee foreign key (employee\_id) references zomato\_employee(employee\_id)

insert into order\_detail(order\_id, customer\_id, restaurant\_id, employee\_id, order\_status, order\_time, delivered\_time) values

(7711, 1001, 101, 210111, 'confirmed', '2019-10-24 12:27:39', '2019-10-24 12:58:39'),

(7712, 1002, 102, 210119, 'confirmed', '2019-10-21 18:35:35', '2019-10-21 19:10:11'),

(7713, 1003, 103, 210112, 'confirmed', '2019-10-22 14:27:11', '2019-10-22 15:00:25'),

(7714, 1004, 104, 210120, 'confirmed', '2019-10-20 11:45:39', '2019-10-20 12:15:49'),

(7715, 1005, 105, 210114, 'confirmed', '2019-10-01 20:01:11', '2019-10-01 21:02:34'),

(7716, 1006, 106, 210113, 'confirmed', '2019-10-02 12:11:39', '2019-10-02 12:45:15'),

(7717, 1001, 107, 210122, 'confirmed', '2019-10-03 11:06:59', '2019-10-03 11:49:35'),

(7718, 1008, 101, 210112, 'confirmed', '2019-10-04 13:16:15', '2019-10-04 14:20:55'),

(7719, 1009, 102, 210125, 'confirmed', '2019-10-28 21:36:39', '2019-10-28 21:45:25'),

(7720, 1010, 103, 210113, 'confirmed', '2019-10-12 12:36:31', '2019-10-12 13:01:19'),

(7721, 1011, 104, 210121, 'confirmed', '2019-10-14 12:36:31', '2019-10-14 13:01:19'),

(7722, 1006, 106, 210119, 'confirmed', '2019-10-20 19:11:30', '2019-10-20 19:45:15'),

(7723, 1007, 105, 210114, 'confirmed', '2019-10-03 20:02:24', '2019-10-03 20:41:59'),

(7724, 1004, 106, 210118, 'confirmed', '2019-10-25 20:01:14', '2019-10-25 20:29:44'),

(7725, 1003, 103, 210114, 'confirmed', '2019-10-22 14:27:11', '2019-10-22 15:00:25'),

(7726, 1014, 107, 210123, 'confirmed', '2019-10-09 12:08:31', '2019-10-09 13:01:36'),

(7727, 1014, 102, 210115, 'confirmed', '2019-10-28 21:36:36', '2019-10-28 21:45:25'),

(7728, 1001, 107, 210122, 'confirmed', '2019-10-05 20:25:24', '2019-10-05 20:59:45'),

(7729, 1003, 107, 210112, 'confirmed', '2019-10-27 14:20:11', '2019-10-27 15:01:25'),

(7730, 1016, 106, 210120, 'confirmed', '2019-10-03 12:10:39', '2019-10-03 12:48:10'),

(7731, 1007, 104, 210113, 'confirmed', '2019-10-21 20:30:45', '2019-10-21 20:55:25'),

(7732, 1001, 103, 210117, 'confirmed', '2019-10-25 12:24:59', '2019-10-25 12:58:20'),

(7733, 1008, 101, 210121, 'confirmed', '2019-10-17 14:11:59', '2019-10-17 14:49:15'),

(7734, 1005, 103, 210118, 'confirmed', '2019-10-12 12:36:31', '2019-10-12 13:01:19'),

(7735, 1001, 105, 210116, 'confirmed', '2019-10-19 11:02:10', '2019-10-19 12:05:14'),

(7736, 1010, 103, 210124, 'confirmed', '2019-10-13 12:48:31', '2019-10-13 13:35:19'),

(7737, 1002, 102, 210124, 'confirmed', '2019-10-23 13:35:35', '2019-10-23 14:02:11'),

(7738, 1001, 106, 210115, 'confirmed', '2019-10-16 12:15:47', '2019-10-16 12:58:25'),

(7739, 1008, 101, 210123, 'confirmed', '2019-10-15 15:58:15', '2019-10-15 16:35:52'),

(7740, 1003, 103, 210121, 'confirmed', '2019-10-25 21:07:11', '2019-10-25 21:57:25'),

(7741, 1001, 107, 210117, 'confirmed', '2019-10-26 12:27:20', '2019-10-26 13:03:18'),

(7742, 1008, 101, 210119, 'confirmed', '2019-10-21 18:11:49', '2019-10-21 19:02:11'),

(7743, 1001, 105, 210111, 'confirmed', '2019-10-13 14:02:19', '2019-10-13 14:58:39'),

(7744, 1010, 103, 210118, 'confirmed', '2019-10-23 12:36:31', '2019-10-23 13:01:10'),

(7745, 1001, 105, 210122, 'confirmed', '2019-10-26 18:27:30', '2019-10-26 18:59:58'),

(7746, 1007, 104, 210117, 'confirmed', '2019-10-21 20:30:45', '2019-10-21 20:55:25'),

(7747, 1001, 106, 210120, 'confirmed', '2019-10-29 19:25:02', '2019-10-29 20:18:39'),

(7748, 1014, 107, 210115, 'confirmed', '2019-10-15 20:01:13', '2019-10-15 20:58:47'),

(7749, 1017, 101, 210111, 'confirmed', '2019-09-14 10:23:13', '2019-09-14 10:51:29'),

(7750, 1013, 103, 210121, 'confirmed', '2019-08-13 13:24:38', '2019-08-13 14:08:05'),

(7751, 1017, 105, 210112, 'confirmed', '2019-09-05 19:01:21', '2019-09-05 19:48:53'),

(7752, 1009, 107, 210115, 'confirmed', '2019-08-19 18:30:13', '2019-08-19 19:14:29'),

(7753, 1001, 102, 210119, 'confirmed', '2019-09-22 20:46:07', '2019-09-22 21:22:56');

select \* from order\_detail;

create table payment\_table(

transaction\_id int(8) not null,

order\_id int(4) not null,

payment\_type varchar(20) not null,

payment\_status varchar(10),

constraint pk\_foods primary key (transaction\_id),

constraint fk\_order\_detail foreign key (order\_id) references order\_detail(order\_id)

);

insert into payment\_table(transaction\_id, order\_id, payment\_type, payment\_status) value

(78945631, 7711, 'cod', 'unpaid'),

(78945632, 7712, 'internet banking', 'paid'),

(78945633, 7713, 'internet banking', 'paid'),

(78945634, 7714, 'cod', 'unpaid'),

(78945635, 7715, 'internet banking', 'paid'),

(78945636, 7716, 'internet banking', 'paid'),

(78945637, 7717, 'cod', 'unpaid'),

(78945638, 7718, 'internet banking', 'paid'),

(78945639, 7719, 'internet banking', 'paid'),

(78945640, 7720, 'cod', 'unpaid'),

(78945641, 7721, 'internet banking', 'paid'),

(78945642, 7722, 'internet banking', 'paid'),

(78945643, 7723, 'cod', 'unpaid'),

(78945644, 7724, 'internet banking', 'paid'),

(78945645, 7725, 'internet banking', 'paid'),

(78945646, 7726, 'internet banking', 'paid'),

(78945647, 7727, 'internet banking', 'paid'),

(78945648, 7728, 'cod', 'unpaid'),

(78945649, 7729, 'internet banking', 'paid'),

(78945650, 7730, 'cod', 'unpaid'),

(78945651, 7731, 'internet banking', 'paid'),

(78945652, 7732, 'cod', 'unpaid'),

(78945653, 7733, 'internet banking', 'paid'),

(78945654, 7734, 'internet banking', 'paid'),

(78945655, 7735, 'cod', 'unpaid'),

(78945656, 7736, 'internet banking', 'paid'),

(78945657, 7737, 'cod', 'unpaid'),

(78945658, 7738, 'internet banking', 'paid'),

(78945659, 7739, 'cod', 'unpaid'),

(78945660, 7740, 'internet banking', 'paid'),

(78945661, 7741, 'cod', 'unpaid'),

(78945662, 7742, 'internet banking', 'paid'),

(78945663, 7743, 'cod', 'unpaid'),

(78945664, 7744, 'internet banking', 'paid'),

(78945665, 7745, 'internet banking', 'paid'),

(78945666, 7746, 'cod', 'unpaid'),

(78945667, 7747, 'internet banking', 'paid'),

(78945668, 7748, 'internet banking', 'paid'),

(78945669, 7749, 'internet banking', 'paid'),

(78945670, 7750, 'internet banking', 'paid'),

(78945671, 7751, 'cod', 'unpaid'),

(78945672, 7752, 'internet banking', 'paid'),

(78945673, 7753, 'internet banking', 'paid');

select \* from payment\_table;

drop table if exists order\_food;

create table order\_food(

orderF\_id int not null auto\_increment,

order\_id int(4) not null,

customer\_id int(4) not null,

restaurant\_id int(3) not null,

food\_id int(7) not null,

quantity int(2),

employee\_id int(6) not null,

constraint pk\_order\_food primary key (orderF\_id),

constraint fk\_payment\_table foreign key (order\_id) references payment\_table(order\_id),

constraint fk\_order\_details foreign key (order\_id) references order\_detail(order\_id),

constraint fk\_foods foreign key (food\_id) references foods(food\_id)

);

insert into order\_food (order\_id, customer\_id, restaurant\_id, food\_id, quantity, employee\_id) values

(7711, 1001, 101, 9999411, 1, 210111),

(7711, 1001, 101, 9999420, 1, 210111),

(7712, 1002, 102, 9999412, 1, 210119),

(7713, 1003, 103, 9999413, 5, 210112),

(7714, 1004, 104, 9999414, 1, 210120),

(7714, 1004, 104, 9999420, 1, 210120),

(7715, 1005, 105, 9999415, 1, 210114),

(7716, 1006, 106, 9999416, 1, 210113),

(7716, 1006, 106, 9999420, 1, 210113),

(7717, 1007, 107, 9999417, 1, 210122),

(7718, 1008, 101, 9999418, 1, 210112),

(7718, 1008, 101, 9999420, 1, 210112),

(7719, 1009, 102, 9999419, 2, 210125),

(7720, 1010, 103, 9999420, 1, 210113),

(7721, 1011, 104, 9999421, 2, 210121),

(7721, 1011, 104, 9999420, 1, 210121),

(7722, 1006, 106, 9999417, 1, 210119),

(7723, 1007, 105, 9999416, 1, 210114),

(7724, 1004, 106, 9999411, 2, 210118),

(7724, 1004, 106, 9999420, 1, 210118),

(7725, 1003, 103, 9999413, 4, 210114),

(7726, 1014, 107, 9999416, 1, 210123),

(7727, 1014, 102, 9999419, 1, 210115),

(7728, 1001, 109, 9999419, 1, 210122),

(7728, 1001, 109, 9999411, 1, 210122),

(7728, 1001, 109, 9999420, 1, 210122),

(7729, 1003, 107, 9999417, 1, 210112),

(7730, 1016, 108, 9999411, 2, 210120),

(7730, 1016, 108, 9999420, 1, 210120),

(7731, 1007, 104, 9999412, 1, 210113),

(7732, 1001, 103, 9999418, 1, 210117),

(7733, 1008, 101, 9999414, 2, 210121),

(7734, 1005, 103, 9999420, 3, 210118),

(7735, 1001, 105, 9999417, 1, 210116),

(7735, 1001, 105, 9999413, 3, 210116),

(7736, 1010, 103, 9999422, 2, 210124),

(7737, 1002, 102, 9999419, 1, 210124),

(7738, 1001, 106, 9999411, 2, 210115),

(7739, 1008, 101, 9999417, 1, 210123),

(7740, 1003, 103, 9999411, 2, 210121),

(7741, 1001, 107, 9999422, 2, 210117),

(7741, 1001, 107, 9999420, 1, 210117),

(7742, 1008, 101, 9999419, 2, 210119),

(7743, 1001, 105, 9999419, 2, 210111),

(7743, 1001, 105, 9999420, 1, 210111),

(7744, 1010, 103, 9999415, 1, 210118),

(7745, 1001, 105, 9999411, 1, 210122),

(7745, 1001, 105, 9999420, 1, 210122),

(7745, 1001, 105, 9999422, 1, 210122),

(7746, 1007, 104, 9999412, 1, 210117),

(7747, 1001, 106, 9999412, 1, 210120),

(7747, 1001, 106, 9999420, 1, 210120),

(7748, 1014, 107, 9999416, 1, 210115),

(7749, 1017, 101, 9999416, 1, 210111),

(7749, 1017, 101, 9999420, 1, 210111),

(7750, 1013, 103, 9999416, 1, 210121),

(7751, 1017, 105, 9999416, 1, 210112),

(7751, 1017, 105, 9999420, 1, 210112),

(7752, 1009, 107, 9999416, 1, 210115),

(7753, 1001, 102, 9999416, 1, 210119),

(7753, 1001, 102, 9999420, 1, 210119);

select \* from order\_food;

#---------------Data Analysis------------------#

# 1) Find Total number of Customer

Select Count(1) as Total\_No\_of\_Customer from customer;

# 2) Count of Customer who never ordered

Select customer\_id, customer\_name from customer

where customer\_id Not In (Select customer\_id from order\_detail);

# 3) Total Sales amout of Each Dish in Zomato in Descending

Select f.food\_id,f.food\_name, round(sum(f.price\_per\_unit),2) as Average\_Sales\_of\_Each\_Dishes

from order\_food fo

join foods f on fo.food\_id = f.food\_id

group by f.food\_id order by Average\_Sales\_of\_Each\_Dishes desc;

#4) Top 5 restaurants Based on No\_of\_Order for October Month

Select r.restaurant\_id,r.restaurant\_name,count(od.order\_id) as Total\_OrderReceived from restaurant r

join

order\_detail od on r.restaurant\_id = od.restaurant\_id

where monthname(od.order\_time) = "OCTOBER"

Group by r.restaurant\_id order by Total\_OrderReceived desc limit 5;

#5) Monthly Revenue sales Analysis

Select r.restaurant\_id,r.restaurant\_name,sum(f.price\_per\_unit) as Monthly\_Revenue\_By\_ZomatoSale from order\_food fo

join restaurant r

join foods f join order\_detail od

on r.restaurant\_id = fo.restaurant\_id and fo.food\_id = f.food\_id and od.order\_id=fo.order\_id

where monthname(od.order\_time) = "OCTOBER"

group by r.restaurant\_id order by r.restaurant\_id ;

#6) Find List of Restaurant which has Repeated customers and its no of times of Purchase

with cte1 as (

Select od.restaurant\_id,r.restaurant\_name,od.customer\_id, row\_number() over (partition by od.restaurant\_id,od.customer\_id order by r.restaurant\_id,od.customer\_id) as Ranks from restaurant r join order\_detail od

on r.restaurant\_id=od.restaurant\_id ),

cte2 as (select restaurant\_id,restaurant\_name,customer\_id,count(ranks) as Customer\_Reoccurences from cte1 group by restaurant\_id,restaurant\_name,customer\_id),

cte3 as (Select restaurant\_id,restaurant\_name,customer\_id,max(Customer\_Reoccurences ) as Loyal\_Customer\_Vist from cte2

group by restaurant\_id ,restaurant\_name,customer\_id order by restaurant\_id ,Loyal\_Customer\_Vist desc )

Select restaurant\_id,restaurant\_name,customer\_id as Loyal\_Customer,Loyal\_Customer\_Vist from cte3 order by Loyal\_Customer\_Vist desc Limit 1;

# 7) Month over Month Analysis of Restaurant sales

with cte1 as (Select r.restaurant\_id,r.restaurant\_name, monthname(od.order\_Time) as Months , month(od.order\_Time) as month\_No, sum(f.price\_per\_unit) as Monthly\_Revenue\_By\_Zomato from order\_food fo

join restaurant r

join foods f join order\_detail od

on r.restaurant\_id = fo.restaurant\_id and fo.food\_id = f.food\_id and od.order\_id=fo.order\_id

group by r.restaurant\_id,Months,month\_No order by r.restaurant\_id,month\_No ),

cte2 as (Select \* , row\_number() over (partition by restaurant\_id ),

(case

when Monthly\_Revenue\_By\_Zomato > lag(Monthly\_Revenue\_By\_Zomato ) over ( order by restaurant\_id,restaurant\_name,month\_No ) then concat(Months , "-" ," Has Progressed in Monthly Revenue" )

else

concat(Months , " has " ,Monthly\_Revenue\_By\_Zomato)

end) as Results from cte1

)

select restaurant\_id,restaurant\_name,Months,Results from cte2;

#8) Each Restaurants with Most Sold Food item

with cte1 as

(select r.restaurant\_id,r.restaurant\_name , f.food\_name,count(f.food\_id) as Ordered\_frequency

from restaurant r join foods f join order\_food fo on r.restaurant\_id=fo.restaurant\_id and f.food\_id=fo.food\_id

group by r.restaurant\_id,r.restaurant\_name,f.food\_name

order by r.restaurant\_id,Ordered\_frequency desc),

cte2 as

(select \* , row\_number() over (partition by restaurant\_id,restaurant\_name order by Ordered\_frequency desc) as RowNum from cte1)

(select \* from cte2 where rownum = 1)

;

#9) Restaurant and its Most Ordered items

with cte1 as

(select r.restaurant\_id,r.restaurant\_name , f.food\_name,count(f.food\_id) as Ordered\_frequency

from restaurant r join foods f join order\_food fo on r.restaurant\_id=fo.restaurant\_id and f.food\_id=fo.food\_id

group by r.restaurant\_id,r.restaurant\_name,f.food\_name

order by r.restaurant\_id,Ordered\_frequency desc),

cte2 as

(select \* from cte1)

select restaurant\_id, restaurant\_name, group\_concat(food\_name) as Restaurant\_Spl\_Menu from cte2 group by restaurant\_id, restaurant\_name;