drop database if exists ecommerce;

create database ecommerce;

use ecommerce;

create table supplier(

SUPP\_ID INT primary key,

SUPP\_NAME varchar(50),

SUPP\_CITY varchar(50),

SUPP\_PHONE varchar(50)

);

create table customer(

CUS\_ID INT primary key,

CUS\_NAME VARCHAR(20),

CUS\_PHONE VARCHAR(10),

CUS\_CITY VARCHAR(30),

CUS\_GENDER CHAR

);

create table category(

CAT\_ID INT primary key,

CAT\_NAME VARCHAR(20)

);

drop table if exists product;

create table product(

PRO\_ID INT primary key,

PRO\_NAME VARCHAR(20)

DEFAULT

"Dummy",

PRO\_DESC VARCHAR(60),

CAT\_ID INT REFERENCES category(CAT\_ID)

);

create table supplier\_pricing(

PRICING\_ID INT primary key,

PRO\_ID INT REFERENCES product(PRO\_ID),

SUPP\_ID INT REFERENCES supplier(SUPP\_ID),

SUPP\_PRICE INT

DEFAULT 0

);

create table orders(

ORD\_ID INT primary key,

ORD\_AMOUNT INT

NOT NULL,

ORD\_DATE DATE

NOT NULL,

CUS\_ID INT REFERENCES customer(CUS\_ID),

PRICING\_ID INT REFERENCES supplier\_pricing(PRICING\_ID)

);

create table rating(

RAT\_ID INT primary key,

ORD\_ID INT REFERENCES orders(ORD\_ID),

RAT\_RATSTARS INT

NOT NULL

);

show tables;

INSERT INTO Supplier

VALUES

(1, 'Rajesh Retails', 'Delhi', '1234567890'),

(2, 'Appario Ltd.', 'Mumbai', '2589631470'),

(3, 'Knome products', 'Bangalore', '9785462315'),

(4, 'Bansal Retails', 'Kochi', '8975463285'),

(5, 'Mittal Ltd.', 'Lucknow', '7898456532');

INSERT INTO Customer

VALUES

(1, 'AAKASH', '9999999999', 'DELHI', 'M'),

(2, 'AMAN', '9785463215', 'NOIDA', 'M'),

(3, 'NEHA', '9999999999', 'MUMBAI', 'F'),

(4, 'MEGHA', '9994562399', 'KOLKATA', 'F'),

(5, 'PULKIT', '7895999999', 'LUCKNOW', 'M');

INSERT INTO Category

VALUES

(1, 'BOOKS'),

(2, 'GAMES'),

(3, 'GROCERIES'),

(4, 'ELECTRONICS'),

(5, 'clothes');

INSERT INTO Product

VALUES

(1, 'GTA V', 'Windows 7 and above with i5 processor and 8GB RAM', 2),

(2, 'TSHIRT', 'SIZE-L with Black, Blue and White variations', 5),

(3, 'ROG LAPTOP', 'Windows 10 with 15inch screen, i7 processor, 1TB SSD', 4),

(4, 'OATS', 'Highly Nutritious from Nestle', 3),

(5, 'HARRY POTTER', 'Best Collection of all time by J.K Rowling', 1),

(6, 'MILK 1L', 'Toned MIlk', 3),

(7, 'Boat Earphones', '1.5Meter long Dolby Atmos', 4),

(8, 'Jeans', 'Stretchable Denim Jeans with various sizes and color', 5),

(9, 'Project IGI', 'compatible with windows 7 and above', 2),

(10, 'Hoodie', 'Black GUCCI for 13 yrs and above', 5),

(11, 'Rich Dad Poor Dad', 'Written by Robert Kiyosaki', 1),

(12, 'Train Your Brain', 'By Shireen Stephen', 1);

INSERT INTO Supplier\_pricing

VALUES

(1, 1, 2, 1500),

(2, 3, 5, 30000),

(3, 5, 1, 3000),

(4, 2, 3, 2500),

(5, 4, 1, 1000),

(6, 12, 2, 780),

(7, 12, 4, 789),

(8, 3, 1, 31000),

(9, 1, 5, 1450),

(10, 4, 2, 999),

(11, 7, 3, 549),

(12, 7, 4, 529),

(13, 6, 2, 105),

(14, 6, 1, 99),

(15, 2, 5, 2999),

(16, 5, 2, 2999);

INSERT INTO orders

VALUES

(101, 1500, '2021-10-06', 2, 1),

(102, 1000, '2021-10-12', 3, 5),

(103, 30000, '2021-09-16', 5, 2),

(104, 1500, '2021-10-05', 1, 1),

(105, 3000, '2021-08-16', 4, 3),

(106, 1450, '2021-08-18', 1, 9),

(107, 789, '2021-09-01', 3, 7),

(108, 780, '2021-09-07', 5, 6),

(109, 3000, '2021-09-10', 5, 3),

(110, 2500, '2021-09-10', 2, 4),

(111, 1000, '2021-09-15', 4, 5),

(112, 789, '2021-09-16', 4, 7),

(113, 31000, '2021-09-16', 1, 8),

(114, 1000, '2021-09-16', 3, 5),

(115, 3000, '2021-09-16', 5, 3),

(116, 99, '2021-09-17', 2, 14);

INSERT INTO Rating

VALUES

(1, 101, 4),

(2, 102, 3),

(3, 103, 1),

(4, 104, 2),

(5, 105, 4),

(6, 106, 3),

(7, 107, 4),

(8, 108, 4),

(9, 109, 3),

(10, 110, 5),

(11, 111, 3),

(12, 112, 4),

(13, 113, 2),

(14, 114, 1),

(15, 115, 1),

(16, 116, 0);

show tables;

SELECT \* FROM Supplier;

SELECT \* FROM CUSTOMER;

SELECT \* FROM Category;

SELECT \* FROM Rating;

SELECT \* FROM Supplier\_pricing;

SELECT \* FROM orders;

SELECT \* FROM product;

#--------------------------------------------------------------------------------------------------------------------------

# 4 Display the total number of customers based on gender who have placed individual orders of worth at least Rs.3000

SELECT cus\_Gender, COUNT(DISTINCT Customer.Cus\_ID) AS TotalCustomers

FROM Orders

JOIN Customer ON Orders.Cus\_ID = Customer.Cus\_ID

WHERE Ord\_Amount >= 3000

GROUP BY cus\_Gender

LIMIT 0, 2000;

#-------------------------------------------------------------------------------------------------------------------

# 5 Display all the orders along with pro\_name ordered by a customer having Cus\_Id=2

SELECT o.ord\_id, o.ord\_amount, o.ord\_date, o.cus\_id, o.pricing\_id, sp.pro\_name

FROM `orders` AS o

INNER JOIN (

SELECT sp.pricing\_id, p.\*

FROM product p

INNER JOIN supplier\_pricing sp ON p.pro\_id = sp.pro\_id

) AS sp ON o.pricing\_id = sp.pricing\_id

WHERE o.cus\_id = '2';

#--------------------------------------------------------------------------------------------------

# 6 Display the Supplier details who can supply more than one product.

SELECT \*

FROM Supplier

WHERE Supp\_ID IN (

SELECT Supp\_ID

FROM supplier\_pricing

GROUP BY Supp\_ID

HAVING COUNT(\*) > 4

);

#----------------------------------------------------------------------------------------------------------------------------------------

# 7 Find the least expensive product from each category and print the table with cat\_id, cat\_name, pro\_name and supp\_price of the produc

SELECT C.CAT\_ID, C.CAT\_NAME, P.PRO\_NAME, SP.min\_price

FROM Category C

INNER JOIN product P ON C.CAT\_ID = P.CAT\_ID

INNER JOIN (

SELECT C.CAT\_ID, MIN(SP.SUPP\_PRICE) AS min\_price

FROM Category C

INNER JOIN product P ON C.CAT\_ID = P.CAT\_ID

INNER JOIN supplier\_pricing SP ON P.PRO\_ID = SP.PRO\_ID

GROUP BY C.CAT\_ID

) AS SP ON C.CAT\_ID = SP.CAT\_ID;

#------------------------------------------------------------------------------------------------------------------------------------

# 8.Display the Id and Name of the Product ordered after “2021-10-05”.

SELECT P.pro\_Id, P.pro\_Name

FROM Product P

INNER JOIN Orders O ON P.pro\_Id = O.pricing\_id

INNER JOIN supplier\_pricing sp ON P.pro\_Id = sp.pricing\_id

WHERE O.Ord\_Date > '2021-10-05';

#------------------------------------------------------------------------------------------------------------------

#9 Display ALL customer name and gender whose names start or end with character 'A'

SELECT CUS\_Name, CUS\_Gender

FROM Customer

WHERE CUS\_Name LIKE 'A%' OR CUS\_Name LIKE '%A';

#------------------------------------------------------------------------------------------------------------------------------------------

/\*10) Create a stored procedure to display supplier id, name, Rating(Average rating of all the products sold by every customer) and

Type\_of\_Service. For Type\_of\_Service, If rating =5, print “Excellent Service”,If rating >4 print “Good Service”, If rating >2 print “Average

Service” else print “Poor Service”. Note that there should be one rating per supplier.\*/

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure`()

BEGIN

SELECT report.supp\_id, report.supp\_name, report.Average,

CASE

WHEN report.Average = 5 THEN 'Excellent Service'

WHEN report.Average > 4 THEN 'Good Service'

WHEN report.Average > 2 THEN 'Average Service'

ELSE 'Poor Service'

END AS Type\_of\_Service FROM

(

SELECT final.supp\_id, supplier.supp\_name, final.Average FROM

(

SELECT test2.supp\_id,SUM(test2.rat\_ratstars) / COUNT(test2.rat\_ratstars) AS Average FROM

(

SELECT supplier\_pricing.supp\_id, test.ORD\_ID, test.RAT\_RATSTARS FROM supplier\_pricing

INNER JOIN

(

SELECT `orders`.pricing\_id, rating.ORD\_ID, rating.RAT\_RATSTARS FROM `orders`

INNER JOIN rating

ON rating.`ord\_id` = `orders`.ord\_id

) AS test

ON test.pricing\_id = supplier\_pricing.pricing\_id

) AS test2 GROUP BY supplier\_pricing.supp\_id

) AS final

INNER JOIN supplier

ON final.supp\_id = supplier.supp\_id

) AS report;

END

call ecommerce.new\_procedure();