USE ecommercedb;

CREATE TABLE `customer` (

`CUS\_ID` int NOT NULL,

`CUS\_NAME` varchar(50) NOT NULL,

`CUS\_PHONE` varchar(10) NOT NULL,

`CUS\_CITY` varchar(30) NOT NULL,

`CUS\_GENDER` char(1) DEFAULT NULL,

PRIMARY KEY (`CUS\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

CREATE TABLE `order` (

`ORD\_ID` int NOT NULL,

`ORD\_AMOUNT` int NOT NULL,

`ORD\_DATE` date DEFAULT NULL,

`CUS\_ID` int NOT NULL,

`PRICING\_ID` int NOT NULL,

PRIMARY KEY (`ORD\_ID`),

KEY `CUS\_ID` (`CUS\_ID`),

KEY `PRICING\_ID` (`PRICING\_ID`),

CONSTRAINT `order\_ibfk\_1` FOREIGN KEY (`CUS\_ID`) REFERENCES `customer` (`CUS\_ID`),

CONSTRAINT `order\_ibfk\_2` FOREIGN KEY (`PRICING\_ID`) REFERENCES `supplier\_pricing` (`PRICING\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

CREATE TABLE `product` (

`PRO\_ID` int NOT NULL,

`PRO\_NAME` varchar(20) NOT NULL DEFAULT 'Dummy',

`PRO\_DESC` varchar(60) NOT NULL,

`CAT\_ID` int NOT NULL,

PRIMARY KEY (`PRO\_ID`),

KEY `CAT\_ID` (`CAT\_ID`),

CONSTRAINT `product\_ibfk\_1` FOREIGN KEY (`CAT\_ID`) REFERENCES `category` (`CAT\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

CREATE TABLE `rating` (

`RAT\_ID` int NOT NULL,

`ORD\_ID` int NOT NULL,

`RAT\_RATSTARS` int NOT NULL,

PRIMARY KEY (`RAT\_ID`),

KEY `ORD\_ID` (`ORD\_ID`),

CONSTRAINT `rating\_ibfk\_1` FOREIGN KEY (`ORD\_ID`) REFERENCES `order` (`ORD\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

CREATE TABLE `supplier` (

`supp\_id` int NOT NULL,

`supp\_name` varchar(50) NOT NULL,

`supp\_city` varchar(50) NOT NULL,

`supp\_phone` varchar(15) NOT NULL,

PRIMARY KEY (`supp\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

CREATE TABLE `supplier\_pricing` (

`PRICING\_ID` int NOT NULL,

`PRO\_ID` int NOT NULL,

`SUPP\_ID` int NOT NULL,

`SUPP\_PRICE` int DEFAULT '0',

PRIMARY KEY (`PRICING\_ID`),

KEY `PRO\_ID` (`PRO\_ID`),

KEY `SUPP\_ID` (`SUPP\_ID`),

CONSTRAINT `supplier\_pricing\_ibfk\_1` FOREIGN KEY (`PRO\_ID`) REFERENCES `product` (`PRO\_ID`),

CONSTRAINT `supplier\_pricing\_ibfk\_2` FOREIGN KEY (`SUPP\_ID`) REFERENCES `supplier` (`supp\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

DESC SUPPLIER

INSERT INTO SUPPLIER VALUES(1,'Rajesh Retails','Delhi','1234567890');

INSERT INTO SUPPLIER VALUES(2,'Appario Ltd. ','Mumbai','2589631470');

INSERT INTO SUPPLIER VALUES(3,'Knome products','Banglore','9785462315');

INSERT INTO SUPPLIER VALUES(4,'Bansal Retails ','Kochi','8975463285');

INSERT INTO SUPPLIER VALUES(5,'Mittal Ltd','Lucknow','7898456532');

DESC CUSTOMER;

INSERT INTO CUSTOMER VALUES(1,'AAKASH','9999999999','DELHI','M');

INSERT INTO CUSTOMER VALUES(2,'AMAN','9785463215','NOIDA','M');

INSERT INTO CUSTOMER VALUES(3,'NEHA','9999999999','MUMBAI','F');

INSERT INTO CUSTOMER VALUES(4,'MEGHA','9994562399','KOLKATA','F');

INSERT INTO CUSTOMER VALUES(5,'PULKIT','7895999999','LUCKNOW','M');

DESC CATEGORY

INSERT INTO CATEGORY VALUES(1,'BOOKS');

INSERT INTO CATEGORY VALUES(2,'GAMES');

INSERT INTO CATEGORY VALUES(3,'GROCERIES');

INSERT INTO CATEGORY VALUES(4,'ELECTRONICS');

INSERT INTO CATEGORY VALUES(5,'CLOTHES');

Desc PRODUCT

INSERT INTO PRODUCT VALUES(1,'GTA V ','Windows 7 and above with i5 processor and 8GB RAM','2');

INSERT INTO PRODUCT VALUES(2,'TSHIRT','SIZE-L with Black, Blue and White variations ','5');

INSERT INTO PRODUCT VALUES(3,'ROG LAPTOP','Windows 10 with 15inch screen, i7 processor, 1TB SSD','4');

INSERT INTO PRODUCT VALUES(4,'OATS','Highly Nutritious from Nestle','3');

INSERT INTO PRODUCT VALUES(5,'HARRY POTTER','Best Collection of all time by J.K Rowling ','1');

INSERT INTO PRODUCT VALUES(6,'MILK','1L Toned MIlk','3');

INSERT INTO PRODUCT VALUES(7,'Boat Earphones','1.5Meter long Dolby Atmos','4');

INSERT INTO PRODUCT VALUES(8,'Jeans','Stretchable Denim Jeans with various sizes and color','5');

INSERT INTO PRODUCT VALUES(9,'Project IGI','compatible with windows 7 and above','2');

INSERT INTO PRODUCT VALUES(10,'Hoodie','Black GUCCI for 13 yrs and above','5');

INSERT INTO PRODUCT VALUES(11,'Rich Dad Poor Dad','Written by RObert Kiyosaki','1');

INSERT INTO PRODUCT VALUES(12,'Train Your Brain','By Shireen Stephen','1');

DESC SUPPLIER\_PRICING

INSERT INTO SUPPLIER\_PRICING VALUES(1,1,2,1500);

INSERT INTO SUPPLIER\_PRICING VALUES(2,3,5,30000);

INSERT INTO SUPPLIER\_PRICING VALUES(3,5,1,3000);

INSERT INTO SUPPLIER\_PRICING VALUES(4,2,3,2500);

INSERT INTO SUPPLIER\_PRICING VALUES(5,4,1,1000);

INSERT INTO SUPPLIER\_PRICING VALUES(6,12,2,780);

INSERT INTO SUPPLIER\_PRICING VALUES(7,12,4,789);

INSERT INTO SUPPLIER\_PRICING VALUES(8,3,1,31000);

INSERT INTO SUPPLIER\_PRICING VALUES(9,1,5,1450);

INSERT INTO SUPPLIER\_PRICING VALUES(10,4,2,999);

INSERT INTO SUPPLIER\_PRICING VALUES(11,7,3,549);

INSERT INTO SUPPLIER\_PRICING VALUES(12,7,4,529);

INSERT INTO SUPPLIER\_PRICING VALUES(13,6,2,105);

INSERT INTO SUPPLIER\_PRICING VALUES(14,6,1,99);

INSERT INTO SUPPLIER\_PRICING VALUES(15,2,5,2999);

INSERT INTO SUPPLIER\_PRICING VALUES(16,2,5,2999);

DESC `ORDER`

INSERT INTO `ORDER` VALUES(101,1500,"2021-10-06",2,1);

INSERT INTO `ORDER` VALUES(102,1000,"2021-10-12",3,5);

INSERT INTO `ORDER` VALUES(103,30000,"2021-09-16",5,2);

INSERT INTO `ORDER` VALUES(104,1500,"2021-10-05",1,1);

INSERT INTO `ORDER` VALUES(105,3000,"2021-08-16",4,3);

INSERT INTO `ORDER` VALUES(106,1450,"2021-08-18",1,9);

INSERT INTO `ORDER` VALUES(107,789,"2021-09-01",3,7);

INSERT INTO `ORDER` VALUES(108,780,"2021-09-07",5,6);

INSERT INTO `ORDER` VALUES(109,3000,"2021-09-10",5,3);

INSERT INTO `ORDER` VALUES(110,2500,"2021-09-10",2,4);

INSERT INTO `ORDER` VALUES(111,1000,"2021-09-15",4,5);

INSERT INTO `ORDER` VALUES(112,789,"2021-09-16",4,7);

INSERT INTO `ORDER` VALUES(113,31000,"2021-09-16",1,8);

INSERT INTO `ORDER` VALUES(114,1000,"2021-09-16",3,5);

INSERT INTO `ORDER` VALUES(115,3000,"2021-09-16",5,3);

INSERT INTO `ORDER` VALUES(116,99,"2021-09-17",2,14);

DESC RATING

INSERT INTO RATING VALUES(1,101,4);

INSERT INTO RATING VALUES(2,102,3);

INSERT INTO RATING VALUES(3,103,1);

INSERT INTO RATING VALUES(4,104,2);

INSERT INTO RATING VALUES(5,105,4);

INSERT INTO RATING VALUES(6,106,3);

INSERT INTO RATING VALUES(7,107,4);

INSERT INTO RATING VALUES(8,108,4);

INSERT INTO RATING VALUES(9,109,3);

INSERT INTO RATING VALUES(10,110,5);

INSERT INTO RATING VALUES(11,111,3);

INSERT INTO RATING VALUES(12,112,4);

INSERT INTO RATING VALUES(13,113,2);

INSERT INTO RATING VALUES(14,114,1);

INSERT INTO RATING VALUES(15,115,1);

INSERT INTO RATING VALUES(16,116,0);

-- 1)**Display the total number of customers based on gender who have placed individual orders of worth at least Rs.3000 --**

show tables;

SELECT COUNT(c.cus\_id) as count , c.cus\_gender FROM `order` ord

INNER JOIN customer c ON c.cus\_id = ord.cus\_id

WHERE ord.ord\_amount>=3000

GROUP BY c.cus\_gender

-- 2) **Display all the orders along with product name ordered by a customer having Customer\_Id=2**

SELECT ord.\*, prd.pro\_name, prd.pro\_desc FROM `order` ord

INNER JOIN SUPPLIER\_PRICING sp ON sp.pricing\_id = ord.pricing\_id

INNER JOIN PRODUCT prd ON prd.pro\_id = sp.pro\_id

WHERE ord.cus\_id = 2

-- 3)**Display the Supplier details who can supply more than one product.**

SELECT \* FROM SUPPLIER sp

INNER JOIN

(

SELECT SUPP\_ID, COUNT(PRO\_ID) as count

FROM SUPPLIER\_PRICING

GROUP BY SUPP\_ID

) as sup ON sup.SUPP\_ID=sp.supp\_id

WHERE sup.count>1

-- 4)**Find the least expensive product from each category and print the table with category id, name, product name and price of the product**

SELECT c.cat\_id,c.cat\_name,MIN(t1.minPrice) FROM category c

INNER JOIN

(

SELECT prd.cat\_id, prd.pro\_name, t2.\* FROM PRODUCT prd

INNER JOIN(

SELECT sp.pro\_id,MIN(sp.supp\_price) as minPrice FROM SUPPLIER\_PRICING sp

GROUP BY sp.pro\_id

) as t2 ON prd.pro\_id=t2.pro\_id

) as t1 ON t1.cat\_id=c.cat\_id

GROUP BY t1.cat\_id

-- 5)**Display the Id and Name of the Product ordered after “2021-10-05”.-- --**

SELECT p.pro\_id,p.pro\_name FROM `order` ord

INNER JOIN SUPPLIER\_PRICING sp ON sp.pricing\_id=ord.pricing\_id

INNER JOIN PRODUCT p ON p.pro\_id = sp.pro\_id

WHERE ord.ord\_date>'2021-10-05'

-- 6)**Display customer name and gender whose names start or end with character 'A'**

SELECT cus.cus\_name, cus.cus\_gender FROM CUSTOMER cus

WHERE cus.cus\_name LIKE 'A%' OR cus.cus\_name like '%A'

-- 7)**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 PROCEDURE details()

BEGIN

SELECT report.supp\_id

,report.supp\_name

,report.rating

,CASE

WHEN report.rating>4 THEN 'Genuine Supplier'

WHEN report.rating>2 THEN 'Average Supplier'

ELSE

'Supplier should not be considered'

END as type\_of\_supplier

FROM

(

SELECT s.supp\_id,s.supp\_name,v.avg as rating FROM SUPPLIER s

INNER JOIN

(

SELECT sp.supp\_id, AVG(rt.rat\_ratstars) as avg

from `order` ord

inner join rating rt ON rt.ord\_id = ord.ord\_id

inner join SUPPLIER\_PRICING sp ON sp.pricing\_id = ord.pricing\_id

GROUP BY sp.supp\_id

)as v ON v.supp\_id = s.supp\_id

) as report;

END

call details()