|  |
| --- |
| Create Database if not exists E\_Commerce; |
| use E\_Commerce; |
| **1.**  Create Table if not exists Supplier(  SUPP\_ID int not null, SUPP\_NAME varchar(50) not null, SUPP\_CITY varchar(50) not null, SUPP\_PHONE varchar(50) not null, primary key (SUPP\_ID)); |
| Create Table if not exists Customer(  CUS\_ID int not null, CUS\_NAME varchar(20) not null, CUS\_PHONE varchar(10) not null, CUS\_CITY varchar(30) not null, CUS\_GENDER char, primary key (CUS\_ID)); |
| Create Table if not exists Category (  CAT\_ID int not null, CAT\_NAME varchar(20) not null, primary key (CAT\_ID)); |
| Create Table if not exists Product (  PRO\_ID int not null, PRO\_NAME varchar(20) not null default "Dummy", PRO\_DESC varchar(60), CAT\_ID int not null, primary key (PRO\_ID), foreign key (CAT\_ID) references Category(CAT\_ID)); |
| Create Table if not exists 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), foreign key (PRO\_ID) references Product(PRO\_ID), foreign key(SUPP\_ID) references Supplier(SUPP\_ID)); |
| Create Table if not exists Orders(  ORD\_ID int not null,ORD\_AMOUNT int not null,ORD\_DATE date not null,CUS\_ID int not null,PRICING\_ID int not null, primary key (ORD\_ID),foreign key (CUS\_ID) references Customer(CUS\_ID),foreign key(PRICING\_ID) references Supplier\_Pricing(PRICING\_ID)); |
| Create Table if not exists Rating (  RAT\_ID int not null, ORD\_ID int not null,RAT\_RATSTARS int not null,  primary key (RAT\_ID), foreign key (ORD\_ID) references Orders(ORD\_ID)); |
| **2.**  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'); |
| 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'); |
| 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"); |
| 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); |
| 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(16,5,2,2999); |
| INSERT INTO Orders VALUES (101,1500,"2021-10-06",2,1);  INSERT INTO Orders VALUES(102,1000,"2021-10-12",3,5);  INSERT INTO Orders VALUES(103,30000,"2021-09-16",5,2);  INSERT INTO Orders VALUES(104,1500,"2021-10-05",1,1);  INSERT INTO Orders VALUES(105,3000,"2021-08-16",4,3);  INSERT INTO Orders VALUES(106,1450,"2021-08-18",1,9);  INSERT INTO Orders VALUES(107,789,"2021-09-01",3,7);  INSERT INTO Orders VALUES(108,780,"2021-09-07",5,6);  INSERT INTO Orders VALUES(109,3000,"2021-09-10",5,3);  INSERT INTO Orders VALUES(110,2500,"2021-09-10",2,4);  INSERT INTO Orders VALUES(111,1000,"2021-09-15",4,5);  INSERT INTO Orders VALUES(112,789,"2021-09-16",4,7);  INSERT INTO Orders VALUES(113,31000,"2021-09-16",1,8);  INSERT INTO Orders VALUES(114,1000,"2021-09-16",3,5);  INSERT INTO Orders VALUES(115,3000,"2021-09-16",5,3);  INSERT INTO Orders VALUES(116,99,"2021-09-17",2,14); |
| 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); |
| **3.**  select count(t2.cus\_gender) as NoOfCustomers, t2.cus\_gender from  (select t1.cus\_id, t1.cus\_gender, t1.ord\_amount, t1.cus\_name from  (select orders.\*, customer.cus\_gender, customer.cus\_name from orders inner join customer where orders.cus\_id=customer.cus\_id  having orders.ord\_amount>=3000) as t1 group by t1.cus\_id) as t2 group by t2.cus\_gender; |
| **4.**  select product.pro\_name, orders.\* from orders, supplier\_pricing, product  where orders.cus\_id=2 and orders.pricing\_id=supplier\_pricing.pricing\_id and supplier\_pricing.pro\_id=product.pro\_id; |
| **5.**  select supplier.\* from supplier where supplier.supp\_id in(select supp\_id from supplier\_pricing group by supp\_id having count(supp\_id)>1) group by supplier.supp\_id; |
| **6.**  select category.cat\_id,category.cat\_name, min(t3.min\_price) as Min\_Price from category inner join  (select product.cat\_id, product.pro\_name, t2.\* from product inner join  (select pro\_id, min(supp\_price) as Min\_Price from supplier\_pricing group by pro\_id)  as t2 where t2.pro\_id = product.pro\_id)  as t3 where t3.cat\_id = category.cat\_id group by t3.cat\_id; |
| **7.**  select product.pro\_id,product.pro\_name from orders inner join supplier\_pricing on supplier\_pricing.pricing\_id=orders.pricing\_id inner join product on product.pro\_id=supplier\_pricing.pro\_id where orders.ord\_date>"2021-10-05"; |
| **8.**  select customer.cus\_name,customer.cus\_gender from customer where customer.cus\_name like 'A%' or customer.cus\_name like '%A'; |
| **9.**  DELIMITER &&  CREATE PROCEDURE Proc()  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 where final.supp\_id = supplier.supp\_id) as report;  END &&  DELIMITER ; |
| call Proc(); |