**QUERIES**

1. Find which menu item is ordered the most so that we can increase the price for that item to increase the profit.

From the **orders** relation, identify the ***itemid*** that are repeated the most and find the ***itemname*** for the corresponding***itemid*** from the **menu** relation

SQL> select A.itemid, B.itemname

2 from menu\_order A, menu B

3 where A.itemid=B.itemid

4 group by A.itemid, B.itemid, B.itemname

5 having count(A.itemid)=(select max(mycount)

6 from (select C.itemid, count(C.itemid) mycount

7 from menu\_order C

8 group by C.itemid));

ITEMID ITEMNAME

---------- --------------------

7 veg lasagna

9 schezwan noodles

5 ceasar salad

3 cream of pumpkin

11 sizzling brownie

1. Find which item is ordered the least so that we can remove that menu item from the list.

From the **orders** relation, identify the ***itemid*** that are repeated the least and find the ***itemname*** for the corresponding***itemid*** from the **menu** relation

SQL> select A.itemid, B.itemname

2 from menu\_order A, menu B

3 where A.itemid=B.itemid

4 group by A.itemid, B.itemid, B.itemname

5 having count(A.itemid)=(select min(mycount)

6 from (select C.itemid, count(C.itemid) mycount

7 from menu\_order C

8 group by C.itemid));

ITEMID ITEMNAME

---------- --------------------

4 vegan split pea

8 mac and cheese

6 pasta salad

1. Find which chefs menu item has the least rating so that we can ask him to replace the menu

From the **orders** relation find the ***itemid*** with the least ***rating***. Find the ***empid*** for the corresponding ***itemid*** from the **menu** relation. Find the employee ***name*** for the corresponding ***empid*** from the **employee** relation.

SQL> select A.itemid, B.itemname, avg(rating) as AvgRating

2 from menu\_order A, menu B

3 where A.itemid=B.itemid

4 group by A.itemid, B.itemname

5 having avg(rating)=(select min(avg(rating))

6 from menu\_order

7 group by itemid);

ITEMID ITEMNAME AVGRATING

---------- -------------------- ----------

6 pasta salad 2

9 schezwan noodles 2

1. Introduce new items as specials for the day and check for the rating, if it has 4 or more out of 5, then add this item to the menu

(*For this functionality, new item will be added as a special. For this, the* ***desc*** *attribute of the* **menu** *relation will have a value called* ***special***). Now for this ***itemid*** , we will check the ***rating*** from the **orders** relation. We will have to check, if this ***rating*** is 4 or more out of 5.

SQL> select A.itemid, B.itemname, avg(rating) as Rating

2 from menu\_order A, menu B

3 where A.itemid=B.itemid

4 and description like '%special%'

5 group by A.itemid, B.itemname

6 having avg(rating)>=4;

ITEMID ITEMNAME RATING

---------- -------------------- ----------

7 veg lasagna 4

1. Find the most bought item and check its nutritional value to see if people actually prefer healthier options, based on this introduce new items to the menu

From the **orders** relation, identify the ***itemid*** that are repeated the most and find the ***itemname*** for the corresponding***itemid*** from the **menu** relation. Check the ***nutriInfo*** of the **menu** relation for this corresponding ***itemid***

SQL> select A.itemid, B.itemname, B.nutriinfo as Nutrition

2 from menu\_order A, menu B

3 where A.itemid=B.itemid

4 group by A.itemid, B.itemid, B.itemname, B.nutriinfo

5 having count(A.itemid)=(select max(mycount)

6 from (select C.itemid, count(C.itemid) mycount

7 from menu\_order C

8 group by C.itemid));

ITEMID ITEMNAME NUTRITION

---------- -------------------- --------------------------------------------------

3 cream of pumpkin 340 cal

7 veg lasagna 540 cal

11 sizzling brownie 320 cal

5 ceasar salad 120 cal

9 schezwan noodles 620 cal

1. Find the most repeated customers

SQL> select phone

2 from date\_order

3 where phone=(select max(mycount)

4 from (select phone, count(phone) as mycount

5 from date\_order

6 group by phone, order\_date));

no rows selected

1. Find the busiest day of the week

SQL> select order\_date as BUSY\_DAY

2 from date\_order

3 group by order\_date

4 having count(order\_date)=(select max(mycount)

5 from(select order\_date, count(order\_date) mycount

6 from date\_order

7 group by order\_date));

BUSY\_DAY

---------

23-OCT-17

1. Supplier who supplies the most number of products for a particular month. (Here we have taken the month as October)

SQL> select distinct A.supplierid,B.suppliername

2 from supply\_info A, supplier B

3 where A.supplierid=B.supplierid and A.supply\_date like '%%OCT%%' and A.supplierid=(select supplierid

4 from supply\_info

5 group by supplierid

6 having count(supplierid)=(select max(mycount)from(select supplierid,count(supplierid) as mycount

7 from supply\_info

8 group by supplierid)));

SUPPLIERID SUPPLIERNAME

---------- --------------------------------------------------

2 DFW Organic Growing