Mini-Project [DBMS]

Great Learning
POWER AHEAD

TITLE :

Data analysis for sales, delivery, and restaurants.

Submitted by :

Sneha Sanjeevkumar (project no :2)

Problem Definition

- Databases:
    - Sales and Delivery
    - Restaurant
- Problem Statement :

1. Retrieving data from sales and delivery domain of an organization based on the given scenarios.
2. Assessing performances of different restaurants based on different options and analyzing the data based on ratings.
3. Creating triggers for an institute to maintain the backup details of the alumni.

Data Set Description

Sales and Delivery:

- This database comprises data from a business organization's sales and delivery domain that has been collected for a decade.
- There are 5 tables in the database.

Table	Information in the table
cust_dimen	Customer name, Province, Region, Customer segment (types), Cust_id(customer ID)
market_fact	Ord_id(ID of the order), Prod_id(product ID), Ship_id(shipping ID), Cust_id, Sales, Discount, Order Quantity, Profit, Shipping_cost, Product_Base_Margin
orders_dimen	Order_id, Order_date, Order_priority, Ord_id
prod_dimen	Product_Category, Product_Sub_Category, Prod_id
shipping_dimen	Order_ID, Ship_Mode, Ship_Date, Ship_id

Restaurant database:

- This database consists of data related to restaurants and users.

Table	Information in the table
chefmozaccepts	placeID, Rpayment
chefmozcuisine	placeID, Rcuisine
chefmozhours4	placeID, hours, days
chefmozparking	placeID, parking_lot
geoplaces2	placeID, latitude, longitude, the_geom_meter, name, address, city, state, country, fax, zip, alcohol, smoking_area, dress_code, accessibility, price, url, Rambience, franchise, area, other_services
rating_final	user_id, placeID, rating, food_rating, service_rating
usercuisine	user_id, Rcuisine
userpayment	user_id, Upayment
userprofile	user_id, latitude, longitude, smoker, drink_level, dress_preference, ambience, transport, marital_status, hijos, birth_year, interest, personality, religion, activity, color, weight, budget, height

Business Importance of Problem

Sales and delivery datasets:

- Tracks sales performances.
- Improve sales performances.
- Helps in organizational decision making.
- Identifying target customers becomes easier.
- Helps to bring in more customers.

Business Importance of Problem

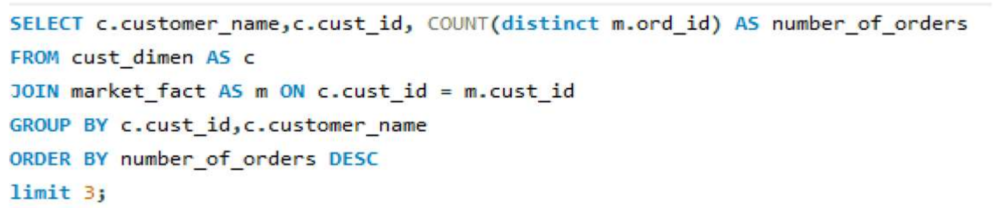
Restaurant datasets:

- Useful in enhancing customer experience.
- Helps in improving the business based on the rating results and new trends.
- Useful in attracting more customers.
- Helps identifying customer preferences.
- Identifies the most sold food.
- Useful in improving the cuisine.

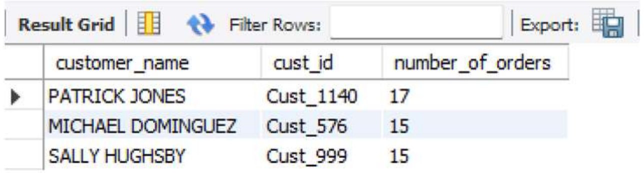
Project Flow – Question 1

Q:Find the top 3 customers who have the maximum number of orders

- First we need to know the top customers with maximum number of orders then extract top 3.



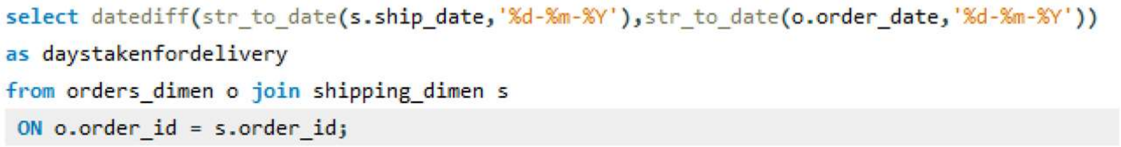
```
SELECT c.customer_name,c.cust_id, COUNT(distinct m.ord_id) AS number_of_orders
FROM cust_dimen AS c
JOIN market_fact AS m ON c.cust_id = m.cust_id
GROUP BY c.cust_id,c.customer_name
ORDER BY number_of_orders DESC
limit 3;
```



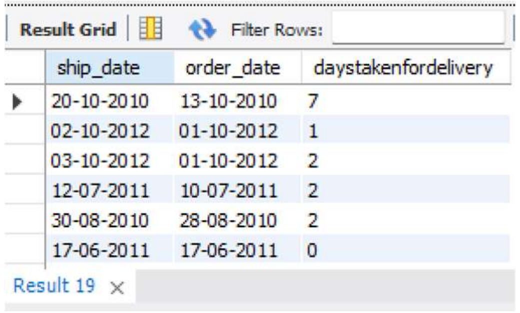
customer_name	cust_id	number_of_orders
PATRICK JONES	Cust_1140	17
MICHAEL DOMINGUEZ	Cust_576	15
SALLY HUGHSBY	Cust_999	15

Project Flow – Question 2

Q: Create a new column DaysTakenForDelivery that contains the date difference between Order_Date and Ship_Date.

- Create a new column 'DaysTakenForDelivery' which specifies the date difference between the 'Order_Date' and the 'Ship_Date'.
- 

```
select datediff(str_to_date(s.ship_date,'%d-%m-%Y'),str_to_date(o.order_date,'%d-%m-%Y'))
as daystakenfordelivery
from orders_dimen o join shipping_dimen s
 ON o.order_id = s.order_id;
```

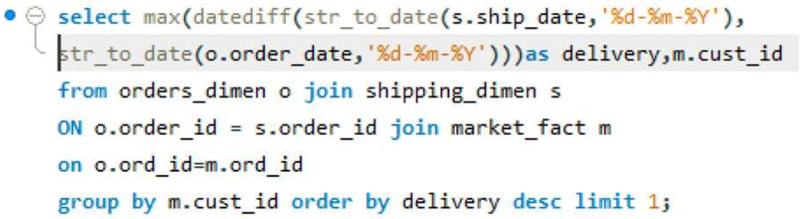


ship_date	order_date	daystakenfordelivery
20-10-2010	13-10-2010	7
02-10-2012	01-10-2012	1
03-10-2012	01-10-2012	2
12-07-2011	10-07-2011	2
30-08-2010	28-08-2010	2
17-06-2011	17-06-2011	0

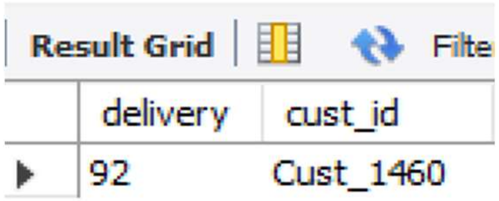
Project Flow – Question 3

Q: Find the customer whose order took the maximum time to get delivered.

- Find the customer whose date difference between order date and ship date is maximum.



```
select max(datediff(str_to_date(s.ship_date,'%d-%m-%Y'),
str_to_date(o.order_date,'%d-%m-%Y')))as delivery,m.cust_id
from orders_dimen o join shipping_dimen s
ON o.order_id = s.order_id join market_fact m
on o.ord_id=m.ord_id
group by m.cust_id order by delivery desc limit 1;
```

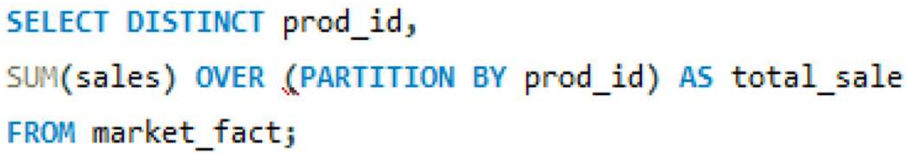


delivery	cust_id
92	Cust_1460

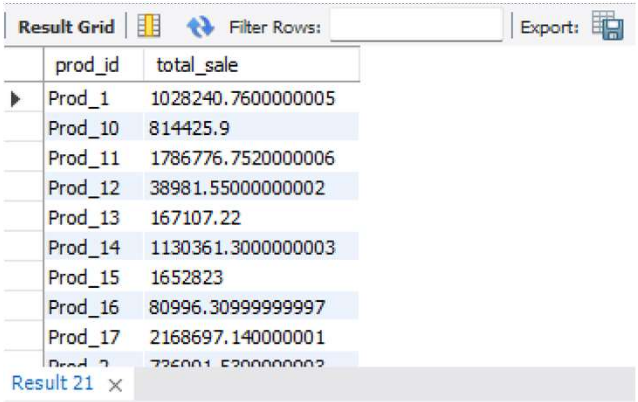
Project Flow – Question 4

Q:Retrieve total sales made by each product from the data (use Windows function)

- Retrieve the sum of total_sales and distinct each product (using the Windows function).



```
SELECT DISTINCT prod_id,
SUM(sales) OVER (PARTITION BY prod_id) AS total_sale
FROM market_fact;
```

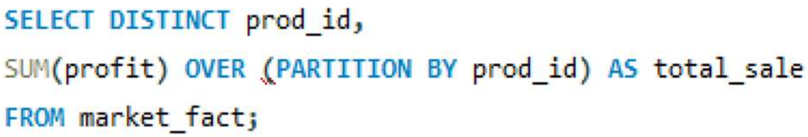


prod_id	total_sale
Prod_1	1028240.7600000005
Prod_10	814425.9
Prod_11	1786776.7520000006
Prod_12	38981.55000000002
Prod_13	167107.22
Prod_14	1130361.3000000003
Prod_15	1652823
Prod_16	80996.30999999997
Prod_17	2168697.140000001
Prod_2	736001.5300000003

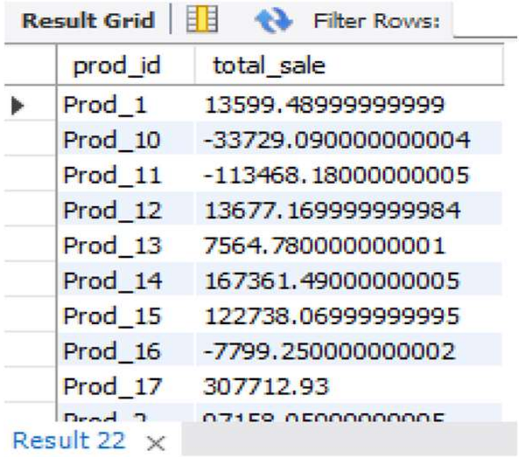
Project Flow – Question 5

Q:Retrieve the total profit made from each product from the data (use windows function)

- Using the data, calculate the sum of profit made from each product (using the Windows function).



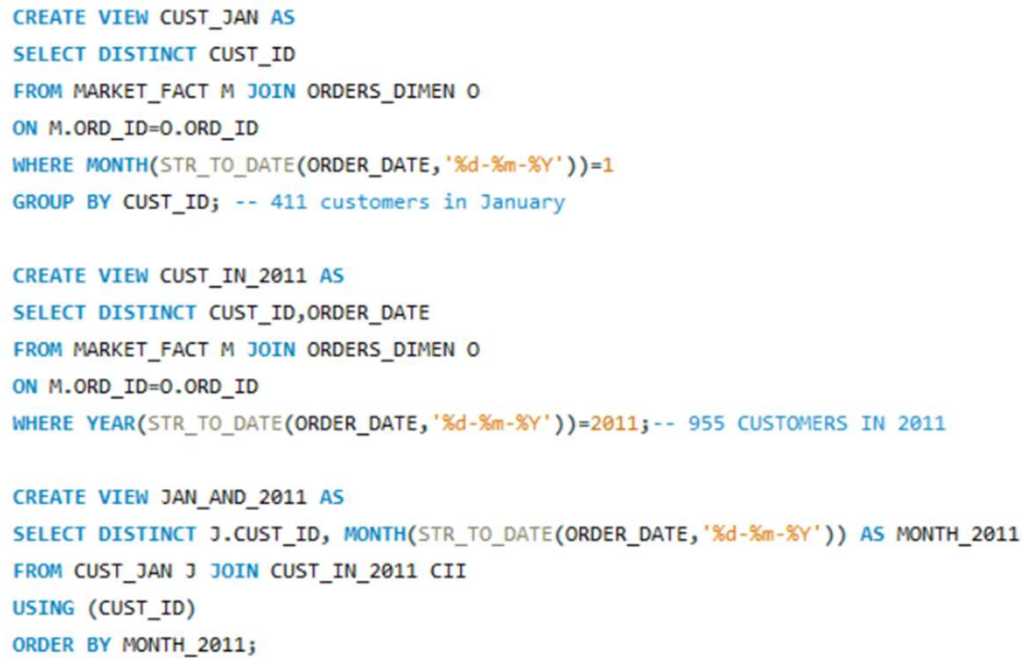
```
SELECT DISTINCT prod_id,
SUM(profit) OVER (PARTITION BY prod_id) AS total_sale
FROM market_fact;
```



prod_id	total_sale
Prod_1	13599.48999999999
Prod_10	-33729.090000000004
Prod_11	-113468.18000000005
Prod_12	13677.169999999984
Prod_13	7564.780000000001
Prod_14	167361.49000000005
Prod_15	122738.06999999995
Prod_16	-7799.250000000002
Prod_17	307712.93
Prod_2	97158.050000000005

Project Flow – Question 6

Q:Count the total number of unique customers in January and how many of them came back every month over the entire year in 2011

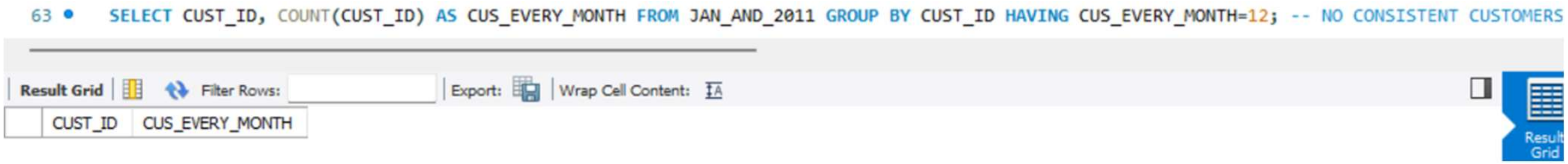


```
CREATE VIEW CUST_JAN AS
SELECT DISTINCT CUST_ID
FROM MARKET_FACT M JOIN ORDERS_DIMEN O
ON M.ORD_ID=O.ORD_ID
WHERE MONTH(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'))=1
GROUP BY CUST_ID; -- 411 customers in January

CREATE VIEW CUST_IN_2011 AS
SELECT DISTINCT CUST_ID,ORDER_DATE
FROM MARKET_FACT M JOIN ORDERS_DIMEN O
ON M.ORD_ID=O.ORD_ID
WHERE YEAR(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'))=2011;-- 955 CUSTOMERS IN 2011

CREATE VIEW JAN_AND_2011 AS
SELECT DISTINCT J.CUST_ID, MONTH(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y')) AS MONTH_2011
FROM CUST_JAN J JOIN CUST_IN_2011 CII
USING (CUST_ID)
ORDER BY MONTH_2011;
```

Project Flow – Question 6

2 views have been created for retrieving customers in January of every year (CUST_JAN) and customers of 2011 (CUST_IN_2011). A third view (JAN_AND_2011) was created from the previous views to obtain the common customer IDs along with the purchase months in 2011. For each month, the customer ID will be present only once. To get the customers who have purchased in all the months of 2011, individual customer IDs are counted and the count should be equal to 12.

Inference: None of the customer have made purchases in all the months of 2011



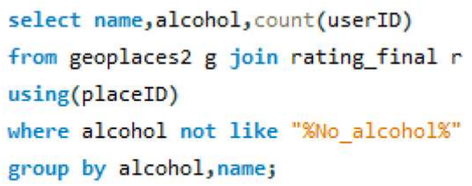
```sql
SELECT CUST_ID, COUNT(CUST_ID) AS CUS_EVERY_MONTH FROM JAN_AND_2011 GROUP BY CUST_ID HAVING CUS_EVERY_MONTH=12; -- NO CONSISTENT CUSTOMERS
```

CUST_ID	CUS_EVERY_MONTH

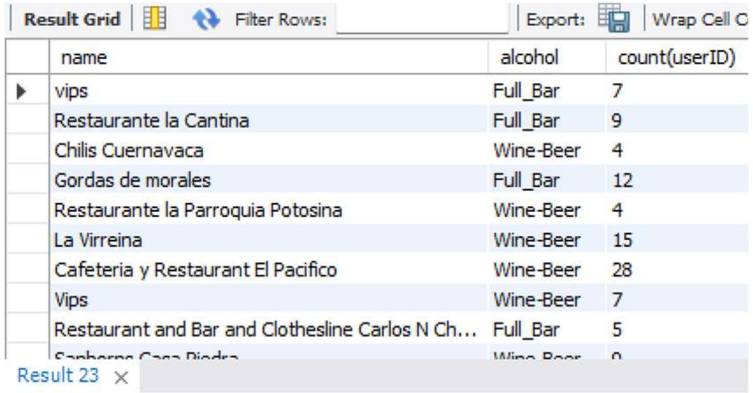
Project Flow – Question 1

Q:We need to find out the total visits to all restaurants under all alcohol categories available.

- We need to know the overall number of visits to all restaurants across all alcohol categories.



```
select name,alcohol,count(userID)
from geoplaces2 g join rating_final r
using(placeID)
where alcohol not like "%No_alcohol%"
group by alcohol,name;
```

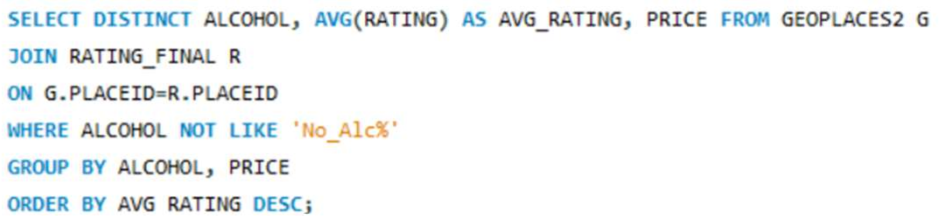


name	alcohol	count(userID)
vips	Full_Bar	7
Restaurante la Cantina	Full_Bar	9
Chilis Cuernavaca	Wine-Beer	4
Gordas de morales	Full_Bar	12
Restaurante la Parroquia Potosina	Wine-Beer	4
La Virreina	Wine-Beer	15
Cafeteria y Restaurant El Pacifico	Wine-Beer	28
Vips	Wine-Beer	7
Restaurant and Bar and Clothesline Carlos N Ch...	Full_Bar	5
Sanborns Casa Piedra	Wine-Beer	9

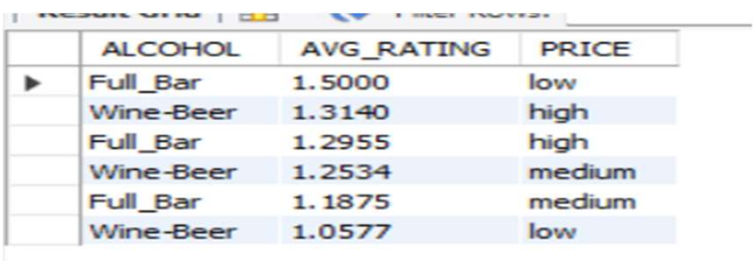
Project Flow – Question 2

Q:-Let's find out the average rating according to alcohol and price so that we can understand the rating in respective price categories as well.

- Let's calculate the average rating based on alcohol and price so that we can comprehend the rating in different pricing ranges.



```
SELECT DISTINCT ALCOHOL, AVG(RATING) AS AVG_RATING, PRICE FROM GEOPLACES2 G
JOIN RATING_FINAL R
ON G.PLACEID=R.PLACEID
WHERE ALCOHOL NOT LIKE 'No_Alc%'
GROUP BY ALCOHOL, PRICE
ORDER BY AVG RATING DESC;
```

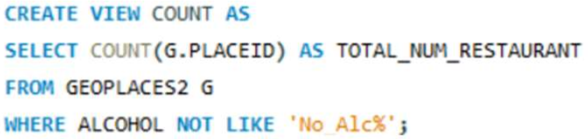


	ALCOHOL	AVG_RATING	PRICE
▶	Full_Bar	1.5000	low
	Wine-Beer	1.3140	high
	Full_Bar	1.2955	high
	Wine-Beer	1.2534	medium
	Full_Bar	1.1875	medium
	Wine-Beer	1.0577	low

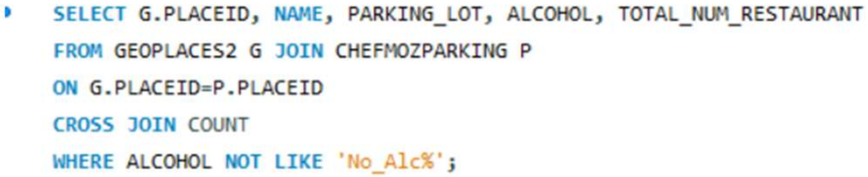
Project Flow – Question 3

Q:Let’s write a query to quantify that what are the parking availability as well in different alcohol categories along with the total number of restaurants.

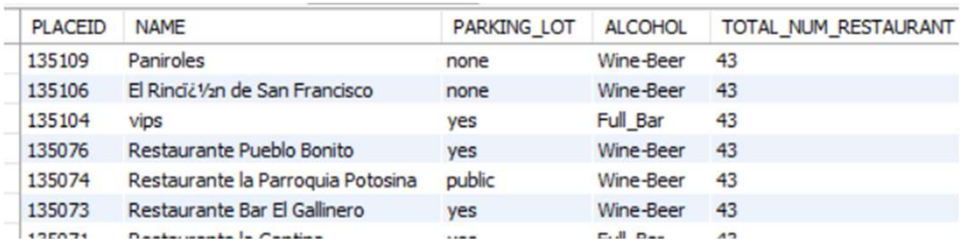
- Let's create a query to determine the availability of parking in various alcohol categories, as well as the overall number of restaurants.



```sql
CREATE VIEW COUNT AS
SELECT COUNT(G.PLACEID) AS TOTAL_NUM_RESTAURANT
FROM GEOPLACES2 G
WHERE ALCOHOL NOT LIKE 'No_Alc%';
```



```sql
SELECT G.PLACEID, NAME, PARKING_LOT, ALCOHOL, TOTAL_NUM_RESTAURANT
FROM GEOPLACES2 G JOIN CHEFMOZPARKING P
ON G.PLACEID=P.PLACEID
CROSS JOIN COUNT
WHERE ALCOHOL NOT LIKE 'No_Alc%';
```

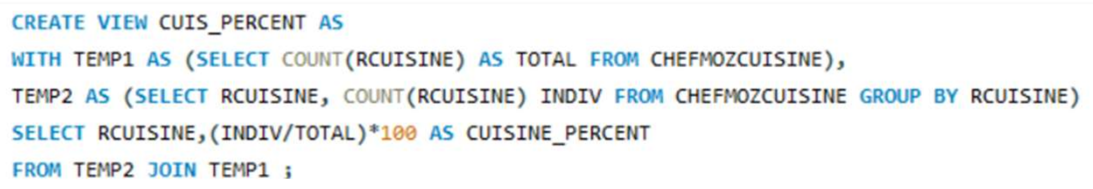


PLACEID	NAME	PARKING_LOT	ALCOHOL	TOTAL_NUM_RESTAURANT
135109	Paniroles	none	Wine-Beer	43
135106	El Rincï¿½n de San Francisco	none	Wine-Beer	43
135104	vips	yes	Full_Bar	43
135076	Restaurante Pueblo Bonito	yes	Wine-Beer	43
135074	Restaurante la Parroquia Potosina	public	Wine-Beer	43
135073	Restaurante Bar El Gallinero	yes	Wine-Beer	43
135071	Restaurante la Cantina	yes	Full_Bar	43

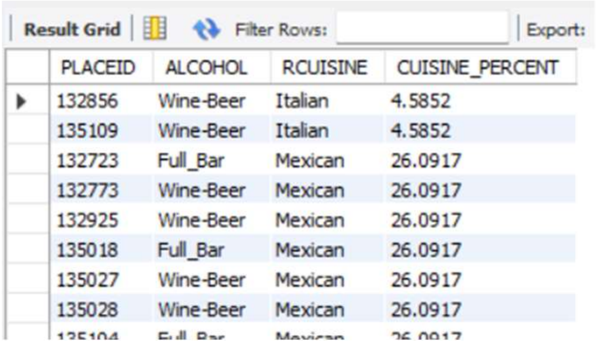
Project Flow – Question 4

Q:Also take out the percentage of different cuisine in each alcohol type.

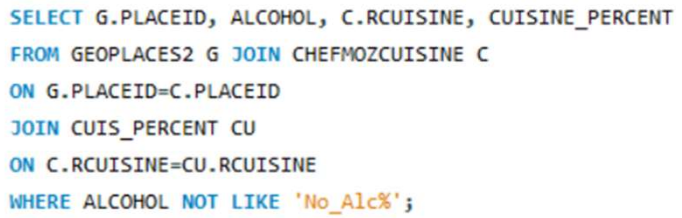
- -Additionally, calculate the percentage of different cuisines in each alcohol type.



```
CREATE VIEW CUIS_PERCENT AS
WITH TEMP1 AS (SELECT COUNT(RCUISINE) AS TOTAL FROM CHEFMOZCUISINE),
TEMP2 AS (SELECT RCUISINE, COUNT(RCUISINE) INDIV FROM CHEFMOZCUISINE GROUP BY RCUISINE)
SELECT RCUISINE,(INDIV/TOTAL)*100 AS CUISINE_PERCENT
FROM TEMP2 JOIN TEMP1 ;
```



PLACEID	ALCOHOL	RCUISINE	CUISINE_PERCENT
132856	Wine-Beer	Italian	4.5852
135109	Wine-Beer	Italian	4.5852
132723	Full_Bar	Mexican	26.0917
132773	Wine-Beer	Mexican	26.0917
132925	Wine-Beer	Mexican	26.0917
135018	Full_Bar	Mexican	26.0917
135027	Wine-Beer	Mexican	26.0917
135028	Wine-Beer	Mexican	26.0917
135104	Full_Bar	Mexican	26.0917

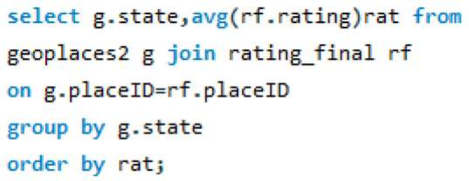


```
SELECT G.PLACEID, ALCOHOL, C.RCUISINE, CUISINE_PERCENT
FROM GEOPLACES2 G JOIN CHEFMOZCUISINE C
ON G.PLACEID=C.PLACEID
JOIN CUIS_PERCENT CU
ON C.RCUISINE=CU.RCUISINE
WHERE ALCOHOL NOT LIKE 'No_Alc%';
```

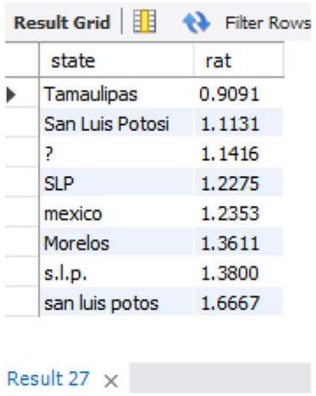
Project Flow – Question 5

Q:let’s take out the average rating of each state.

- Take the average rating for each state.



```
select g.state,avg(rf.rating)rat from
geoplaces2 g join rating_final rf
on g.placeID=rf.placeID
group by g.state
order by rat;
```

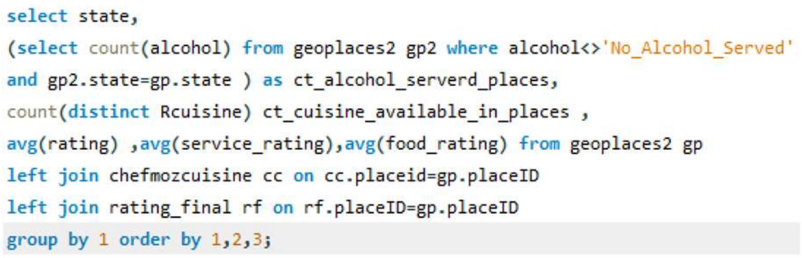


state	rat
Tamaulipas	0.9091
San Luis Potosi	1.1131
?	1.1416
SLP	1.2275
mexico	1.2353
Morelos	1.3611
s.l.p.	1.3800
san luis potos	1.6667

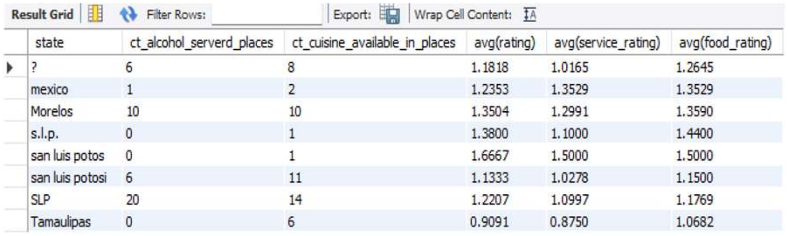
Project Flow – Question 6

Q:' Tamaulipas' Is the lowest average rated state. Quantify the reason why it is the lowest rated by providing the summary on the basis of State, alcohol, and Cuisine.

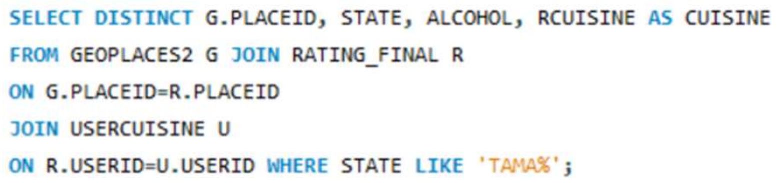
- Though Tamaulipa serves a variety of cuisines, Tamaulipa restaurants never served alcohol



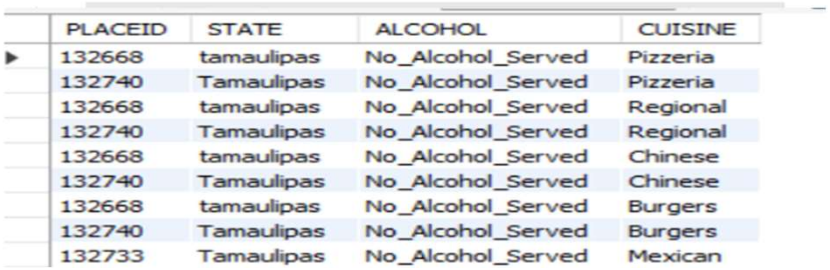
```
select state,
(select count(alcohol) from geoplaces2 gp2 where alcohol<>'No_Alcohol_Served'
and gp2.state=gp.state ) as ct_alcohol_serverd_places,
count(distinct Rcuisine) ct_cuisine_available_in_places ,
avg(rating) ,avg(service_rating),avg(food_rating) from geoplaces2 gp
left join chefmozcuisine cc on cc.placeid=gp.placeID
left join rating_final rf on rf.placeID=gp.placeID
group by 1 order by 1,2,3;
```



state	ct_alcohol_serverd_places	ct_cuisine_available_in_places	avg(rating)	avg(service_rating)	avg(food_rating)
?	6	8	1.1818	1.0165	1.2645
mexico	1	2	1.2353	1.3529	1.3529
Morelos	10	10	1.3504	1.2991	1.3590
s.l.p.	0	1	1.3800	1.1000	1.4400
san luis potos	0	1	1.6667	1.5000	1.5000
san luis potosi	6	11	1.1333	1.0278	1.1500
SLP	20	14	1.2207	1.0997	1.1769
Tamaulipas	0	6	0.9091	0.8750	1.0682



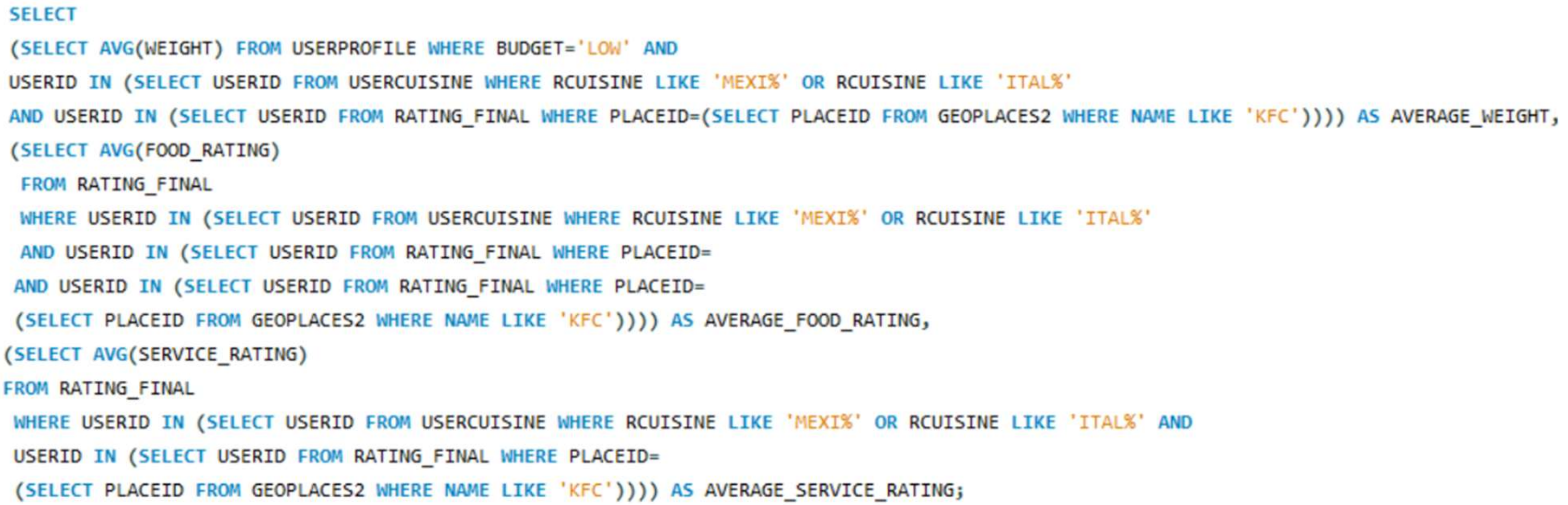
```
SELECT DISTINCT G.PLACEID, STATE, ALCOHOL, RCUISINE AS CUISINE
FROM GEOPLACES2 G JOIN RATING_FINAL R
ON G.PLACEID=R.PLACEID
JOIN USERCUISINE U
ON R.USERID=U.USERID WHERE STATE LIKE 'TAMA%';
```



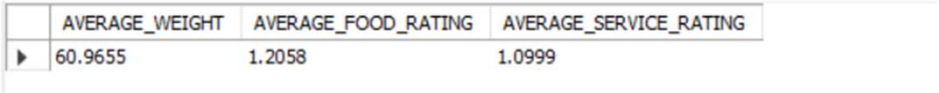
PLACEID	STATE	ALCOHOL	CUISINE
132668	tamaulipas	No_Alcohol_Served	Pizzeria
132740	Tamaulipas	No_Alcohol_Served	Pizzeria
132668	tamaulipas	No_Alcohol_Served	Regional
132740	Tamaulipas	No_Alcohol_Served	Regional
132668	tamaulipas	No_Alcohol_Served	Chinese
132740	Tamaulipas	No_Alcohol_Served	Chinese
132668	tamaulipas	No_Alcohol_Served	Burgers
132740	Tamaulipas	No_Alcohol_Served	Burgers
132733	Tamaulipas	No_Alcohol_Served	Mexican

Project Flow – Question 7

Q:**Find the average weight, food rating, and service rating of the customers who have visited KFC and tried Mexican or Italian types of cuisine, and also their budget level is low. We encourage you to give it a try by not using joins. –** All the conditions where applied to all the required attributes.



```
SELECT
(SELECT AVG(WEIGHT) FROM USERPROFILE WHERE BUDGET='LOW' AND
USERID IN (SELECT USERID FROM USERCUISINE WHERE RCUISINE LIKE 'MEXI%' OR RCUISINE LIKE 'ITAL%'
AND USERID IN (SELECT USERID FROM RATING_FINAL WHERE PLACEID=(SELECT PLACEID FROM GEOPLACES2 WHERE NAME LIKE 'KFC')))) AS AVERAGE_WEIGHT,
(SELECT AVG(FOOD_RATING)
 FROM RATING_FINAL
 WHERE USERID IN (SELECT USERID FROM USERCUISINE WHERE RCUISINE LIKE 'MEXI%' OR RCUISINE LIKE 'ITAL%'
 AND USERID IN (SELECT USERID FROM RATING_FINAL WHERE PLACEID=
 AND USERID IN (SELECT USERID FROM RATING_FINAL WHERE PLACEID=
 (SELECT PLACEID FROM GEOPLACES2 WHERE NAME LIKE 'KFC')))) AS AVERAGE_FOOD_RATING,
(SELECT AVG(SERVICE_RATING)
FROM RATING_FINAL
 WHERE USERID IN (SELECT USERID FROM USERCUISINE WHERE RCUISINE LIKE 'MEXI%' OR RCUISINE LIKE 'ITAL%' AND
 USERID IN (SELECT USERID FROM RATING_FINAL WHERE PLACEID=
 (SELECT PLACEID FROM GEOPLACES2 WHERE NAME LIKE 'KFC')))) AS AVERAGE_SERVICE_RATING;
```



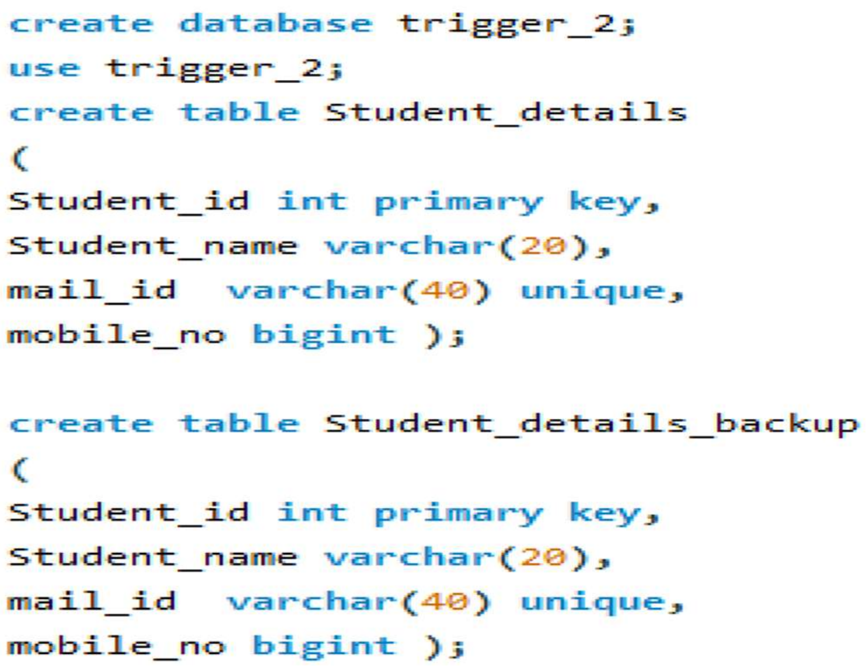
	AVERAGE_WEIGHT	AVERAGE_FOOD_RATING	AVERAGE_SERVICE_RATING
▶	60.9655	1.2058	1.0999

TRIGGER

- Trigger is a statement that a system executes automatically when there is any modification to the database. In a trigger, we first specify when the trigger is to be executed and then the action to be performed when the trigger executes. Triggers are used to specify certain integrity constraints and referential constraints that cannot be specified using the constraint mechanism of SQL.

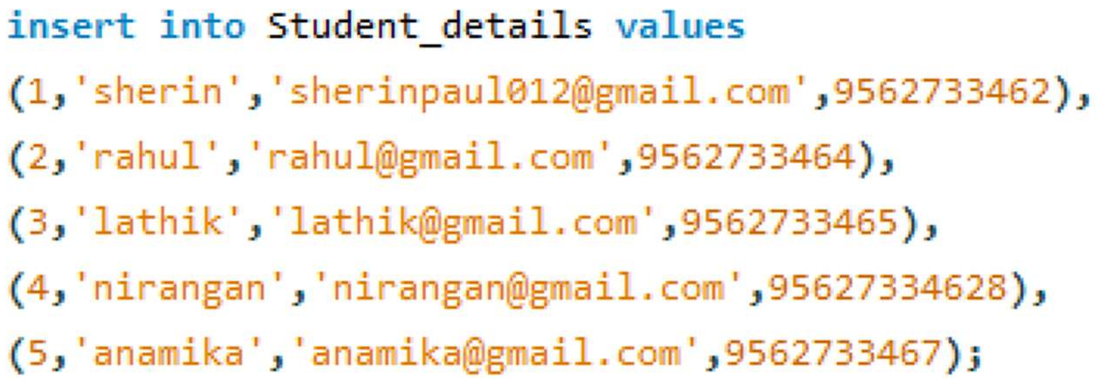
Trigger – Question 1

Q:Create two called Student_details and Student_details_backup.

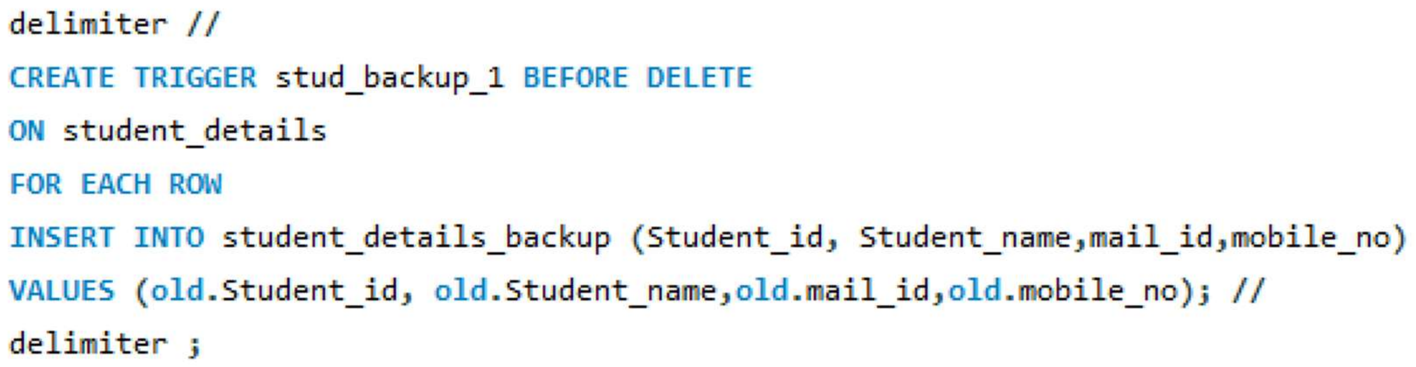


```
create database trigger_2;
use trigger_2;
create table Student_details
(
Student_id int primary key,
Student_name varchar(20),
mail_id  varchar(40) unique,
mobile_no bigint );

create table Student_details_backup
(
Student_id int primary key,
Student_name varchar(20),
mail_id  varchar(40) unique,
mobile_no bigint );
```

Trigger – Question 1



```
insert into Student_details values
(1,'sherin','sherinpaul012@gmail.com',9562733462),
(2,'rahul','rahul@gmail.com',9562733464),
(3,'lathik','lathik@gmail.com',9562733465),
(4,'nirangan','nirangan@gmail.com',95627334628),
(5,'anamika','anamika@gmail.com',9562733467);
```

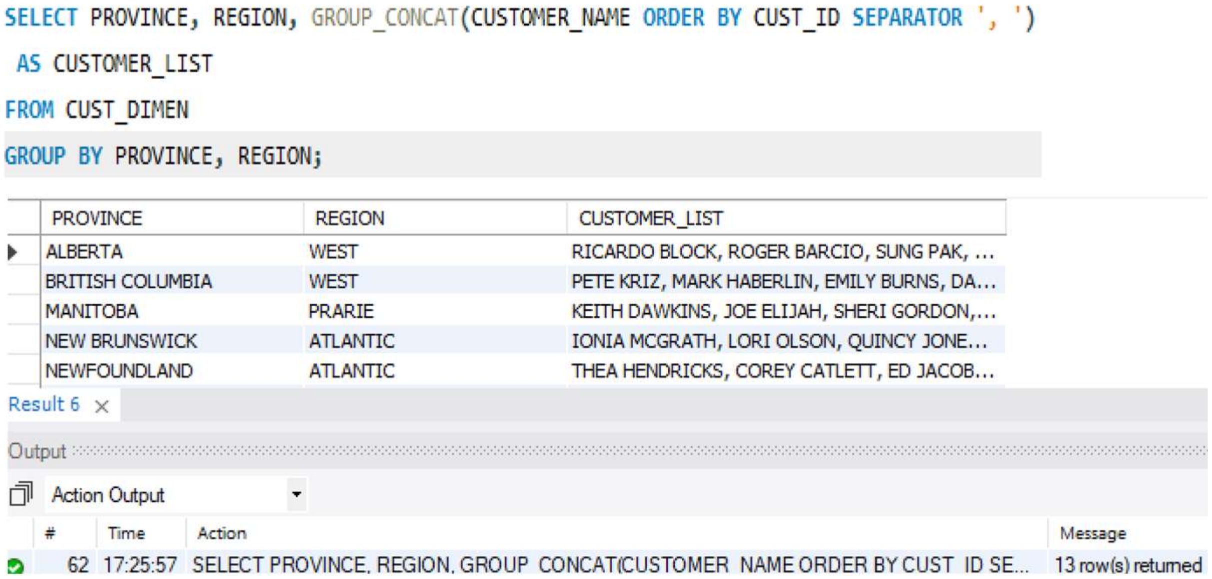


```
delimiter //
CREATE TRIGGER stud_backup_1 BEFORE DELETE
ON student_details
FOR EACH ROW
INSERT INTO student_details_backup (Student_id, Student_name,mail_id,mobile_no)
VALUES (old.Student_id, old.Student_name,old.mail_id,old.mobile_no); //
delimiter ;
```

Major Challenge

Q1: Use of GROUP_CONCAT function to show customers from same city.

GROUP_CONCAT() here is useful displaying all the customers from the same location in a single row.



```
SELECT PROVINCE, REGION, GROUP_CONCAT(CUSTOMER_NAME ORDER BY CUST_ID SEPARATOR ', ')
 AS CUSTOMER_LIST
FROM CUST_DIMEN
GROUP BY PROVINCE, REGION;
```

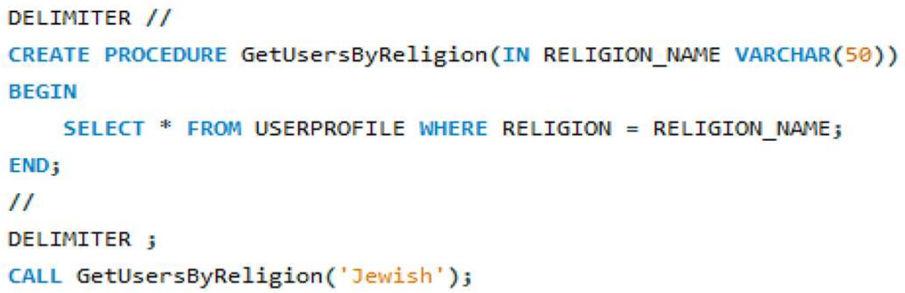
PROVINCE	REGION	CUSTOMER_LIST
ALBERTA	WEST	RICARDO BLOCK, ROGER BARCIO, SUNG PAK, ...
BRITISH COLUMBIA	WEST	PETE KRIZ, MARK HABERLIN, EMILY BURNS, DA...
MANITOBA	PRARIE	KEITH DAWKINS, JOE ELIJAH, SHERI GORDON,...
NEW BRUNSWICK	ATLANTIC	IONIA MCGRATH, LORI OLSON, QUINCY JONE...
NEWFOUNDLAND	ATLANTIC	THEA HENDRICKS, COREY CATLETT, ED JACOB...

```
#    Time      Action                                                              Message
62   17:25:57  SELECT PROVINCE, REGION, GROUP_CONCAT(CUSTOMER_NAME ORDER BY CUST_ID SE...  13 row(s) returned
```

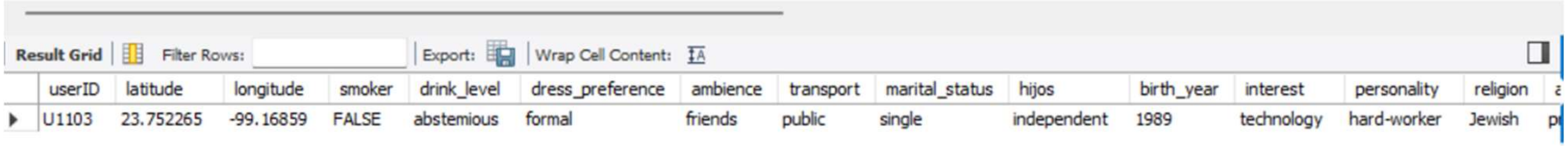
Major Challenge

Q2: Creating stored procedure to retrieve user ids based on religion. The stored procedure will be useful to make attractive offers during festival times.

Creating a stored procedure is useful in applying a set of code whenever required, just by calling the stored procedure's name along with the wanted record name.



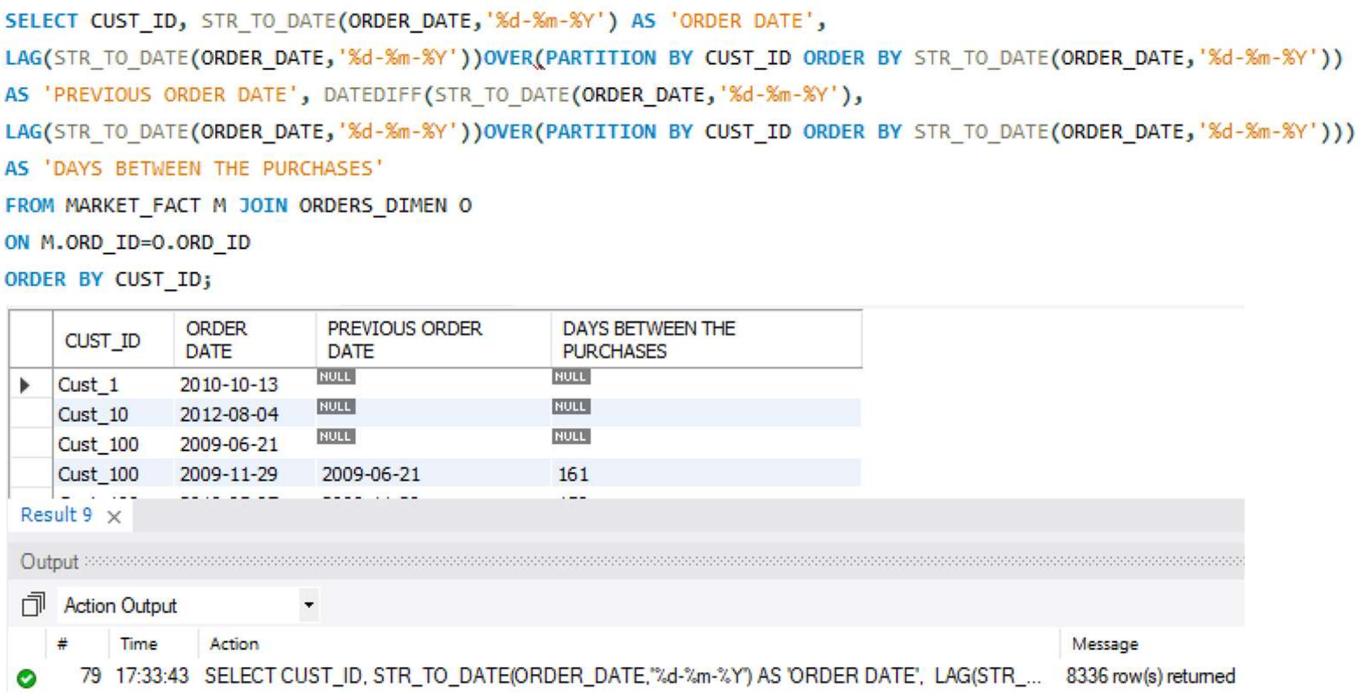
```
DELIMITER //
CREATE PROCEDURE GetUsersByReligion(IN RELIGION_NAME VARCHAR(50))
BEGIN
    SELECT * FROM USERPROFILE WHERE RELIGION = RELIGION_NAME;
END;
//
DELIMITER ;
CALL GetUsersByReligion('Jewish');
```



userID	latitude	longitude	smoker	drink_level	dress_preference	ambience	transport	marital_status	hijos	birth_year	interest	personality	religion	a
U1103	23.752265	-99.16859	FALSE	abstemious	formal	friends	public	single	independent	1989	technology	hard-worker	Jewish	p

Major Challenge

Q3:Calculate delta values of order date for each customer



```
SELECT CUST_ID, STR_TO_DATE(ORDER_DATE,'%d-%m-%Y') AS 'ORDER DATE',
LAG(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'))OVER(PARTITION BY CUST_ID ORDER BY STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'))
AS 'PREVIOUS ORDER DATE', DATEDIFF(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'),
LAG(STR_TO_DATE(ORDER_DATE,'%d-%m-%Y'))OVER(PARTITION BY CUST_ID ORDER BY STR_TO_DATE(ORDER_DATE,'%d-%m-%Y')))
AS 'DAYS BETWEEN THE PURCHASES'
FROM MARKET_FACT M JOIN ORDERS_DIMEN O
ON M.ORD_ID=O.ORD_ID
ORDER BY CUST_ID;
```

CUST_ID	ORDER DATE	PREVIOUS ORDER DATE	DAYS BETWEEN THE PURCHASES
Cust_1	2010-10-13	NULL	NULL
Cust_10	2012-08-04	NULL	NULL
Cust_100	2009-06-21	NULL	NULL
Cust_100	2009-11-29	2009-06-21	161

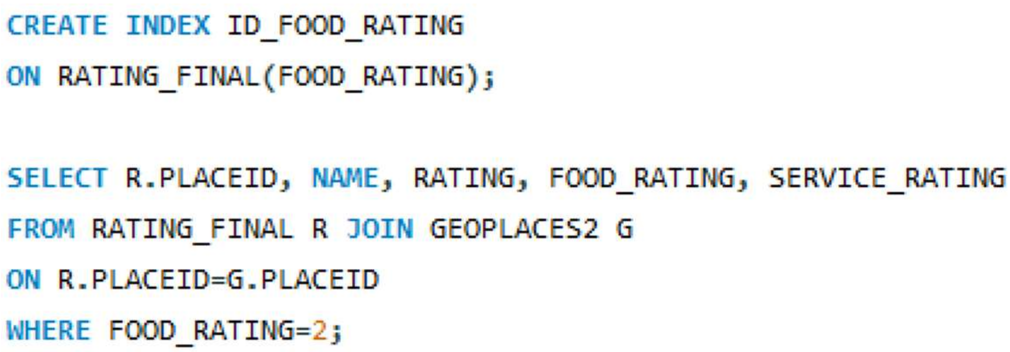
```
#   Time      Action                                                                          Message
79  17:33:43  SELECT CUST_ID, STR_TO_DATE(ORDER_DATE,'%d-%m-%Y') AS 'ORDER DATE',  LAG(STR_...  8336 row(s) returned
```

The purpose of getting delta values between the order dates for each customer is to get the number days between the previous and current purchases.

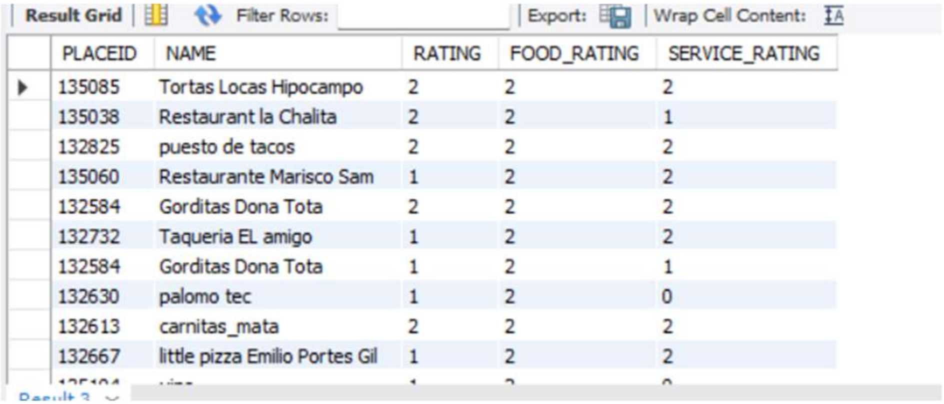
Major Challenge

Q4:Creating index on food_rating to retrieve restaurant information easily.

Assigning indexes on food rating, allows the users to quickly sift through a list of restaurants based on the food rating. Here 2 is the highest rating.



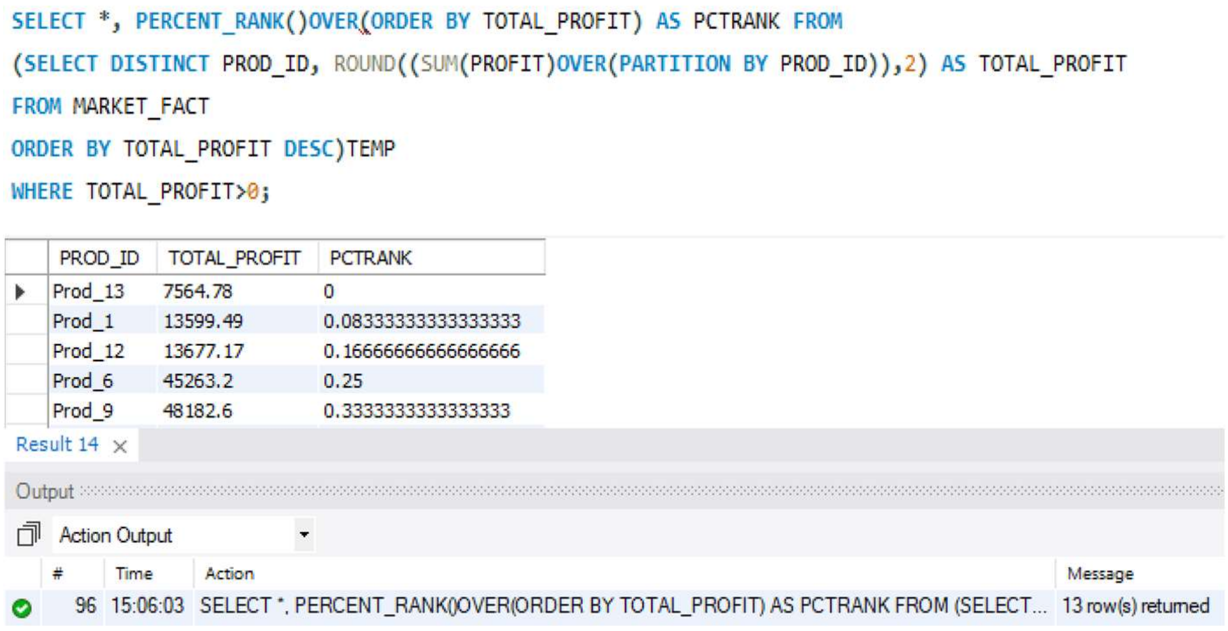
```
CREATE INDEX ID_FOOD_RATING
ON RATING_FINAL(FOOD_RATING);

SELECT R.PLACEID, NAME, RATING, FOOD_RATING, SERVICE_RATING
FROM RATING_FINAL R JOIN GEOPLACES2 G
ON R.PLACEID=G.PLACEID
WHERE FOOD_RATING=2;
```



PLACEID	NAME	RATING	FOOD_RATING	SERVICE_RATING
135085	Tortas Locas Hipocampo	2	2	2
135038	Restaurant la Chalita	2	2	1
132825	puesto de tacos	2	2	2
135060	Restaurante Marisco Sam	1	2	2
132584	Gorditas Dona Tota	2	2	2
132732	Taqueria EL amigo	1	2	2
132584	Gorditas Dona Tota	1	2	1
132630	palomo tec	1	2	0
132613	carnitas_mata	2	2	2
132667	little pizza Emilio Portes Gil	1	2	2
135104	vips	1	2	0

Major Challenge

Q5:Retrieve the total profit made from each product from the data and give percentile ranks where total profit is positive.



```
SELECT *, PERCENT_RANK()OVER(ORDER BY TOTAL_PROFIT) AS PCTRANK FROM
(SELECT DISTINCT PROD_ID, ROUND((SUM(PROFIT)OVER(PARTITION BY PROD_ID)),2) AS TOTAL_PROFIT
FROM MARKET_FACT
ORDER BY TOTAL_PROFIT DESC)TEMP
WHERE TOTAL_PROFIT>0;
```

PROD_ID	TOTAL_PROFIT	PCTRANK
Prod_13	7564.78	0
Prod_1	13599.49	0.08333333333333333
Prod_12	13677.17	0.16666666666666666
Prod_6	45263.2	0.25
Prod_9	48182.6	0.3333333333333333

#	Time	Action	Message
96	15:06:03	SELECT *, PERCENT_RANK()OVER(ORDER BY TOTAL_PROFIT) AS PCTRANK FROM (SELECT...	13 row(s) returned

The products are given percentile ranks between 0 to 1 where the total profit is positive. 0 is given for the lowest total profit and 1 for the highest total profit.

Conclusions

- We got to learn and use triggers, stored procedures, and indexes in MySQL.
- From the datasets of sales and delivery and restaurant, we were able analyze the performances, retrieve data based on certain conditions, analyze customer choices based on profits and ratings, and find the factors that affect the growth of sales in an organization and performances of restaurants.

Thank You !!!....