SQL

Structured Query Language

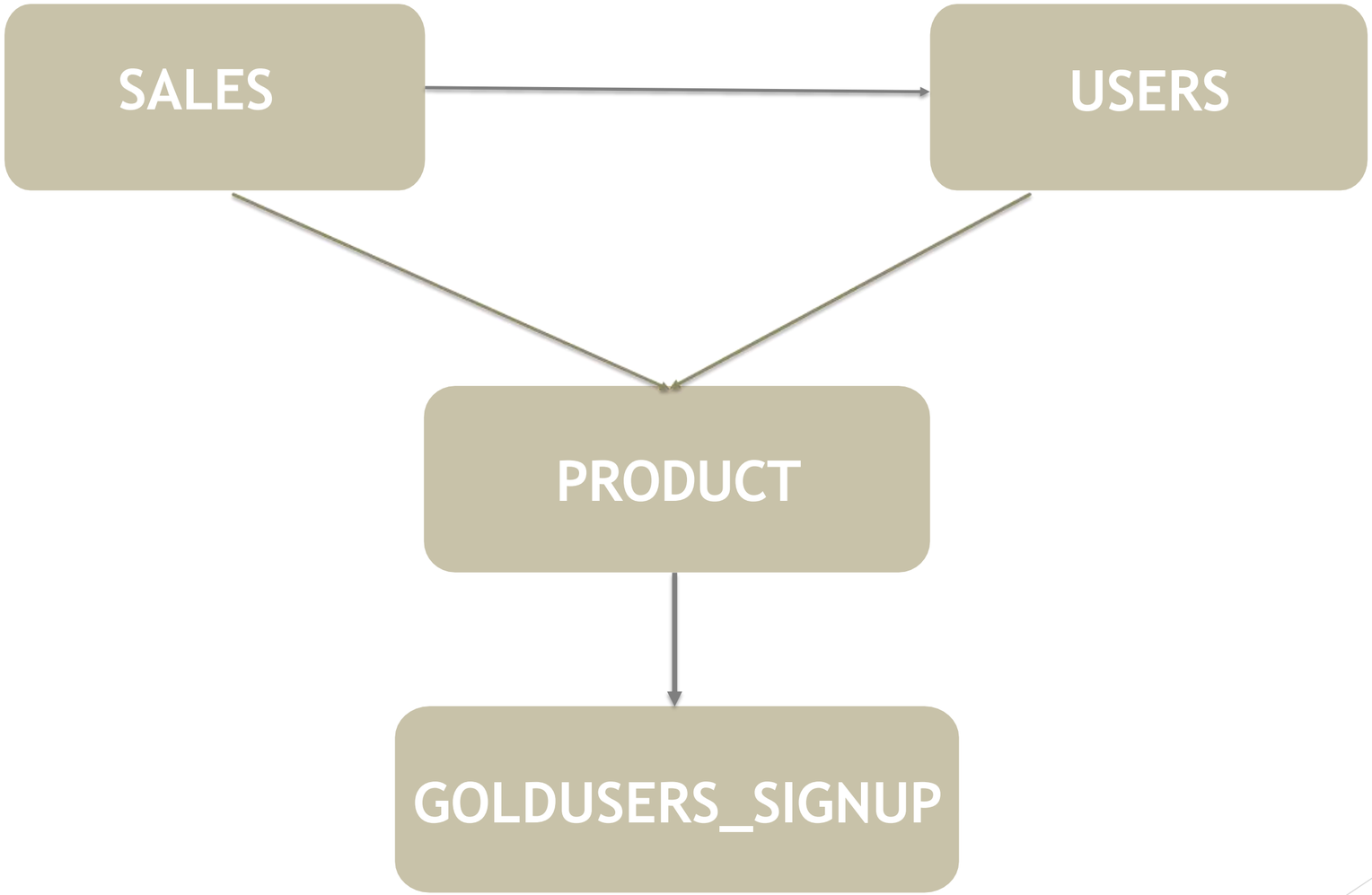
Presented by:
P DILIP KUMAR

Project Title

ZOMATO DATA EXPLORATION

Zomato is an Indian Multinational restaurant aggregator and food delivery company, founded by Deepinder Goyal and Pankaj Chaddah in 2008. Zomato provides information, menus and user-reviews of restaurants as well as food delivery options from partner restaurants in more than 1,000 Indian cities and towns, as of 2022-23.

Entity Relationship Diagram



Sales
Userid
Created_date
Product_id

Users
Userid
Signup_date

Product
Product_id
Product_name
Price

Goldusers_signup
Userid
Gold_signup_date

SALES TABLE

userid	created_date	product_id
1	2017-04-19	2
3	2019-12-18	1
2	2020-07-20	3
1	2019-10-23	2
1	2018-03-19	3
3	2016-12-20	2
1	2016-11-09	1
1	2016-05-20	3
2	2017-09-24	1
1	2017-03-11	2
1	2016-03-11	1
3	2016-10-11	1
3	2017-07-12	2
3	2016-12-15	2
2	2017-08-11	2
2	2018-10-09	3

USERS TABLE

userid	signup_date
1	2014-02-09
2	2015-01-15
3	2014-04-11

PRODUCT TABLE

product_id	product_name	price
1	p1	980
2	p2	870
3	p3	330

GOLDUSERS_SIGNUP TABLE

userid	gold_signup_date
1	2017-09-22
3	2017-04-21

What is the total amount each customer spent on Zomato ?

```
Select s.userid, sum(p.price) as Total_amount_spent from sales s right join
product p on s.product_id=p.product_id group by userid order by userid;
```

OUTPUT

	userid	Total_amount_spent
▶	1	5230
	2	2510
	3	4570

How many days visited each customer on Zomato ?

```
select userid, count(unique_dates)as Distinct_days from
(Select distinct(created_date)as unique_dates, userid
from sales) x group by userid order by userid;
```

OUTPUT

	userid	Distinct_days
▶	1	7
	2	4
	3	5

What was the first product purchased by each customer?

```
select * from(
        select s.userid, p.product_name, s.created_date,
        row_  number() over (partition by  s.userid order by
        s.created_date) as first_product  from sales s
        inner join product p on s.product_id=p.product_id)x
        where first_product =1;
```

OUTPUT

	userid	created_date	product_id	first_product
▶	1	2016-03-11	1	1
	2	2017-08-11	2	1
	3	2016-10-11	1	1

what was the most purchased item on the menu and how many times was it purchased by all customer?

```
select userid, count(product_id) as count from sales where
product_id = (
                    select product_id  from sales group by
                    product_id order by count(product_id)
                    desc limit 1) group by userid;
```

OUTPUT

	userid	count
▶	1	3
	3	3
	2	1

Which item was the most popular for each customer?

```
select * from (
        select *,row_number() over(partition by userid
        order by count desc) as row_no from (
        select userid, product_id, count(product_id) as count
        from sales group by userid,product_id) a ) b
        where row_no=1;
```

OUTPUT

	userid	product_id	count	row_no
▶	1	2	3	1
	2	3	2	1
	3	2	3	1

Which item was purchased first by the customer after they became a gold member?

```
Select * from (
        Select *,row_number() over(partition by userid order by created_date)
        as row_no  from ( Select s.userid, s.product_id, s.created_date from
        sales s inner join goldusers_signup g on s.userid=g.userid
       where s.created__date>g.gold_signup_date) x) y where row_no=1;
```

OUTPUT

	userid	product_id	created_date	gold_signup_date	row_no
▶	1	3	2018-03-19	2017-09-22	1
	3	2	2017-07-12	2017-04-21	1

Which item was purchased just before the customer became a gold member?

```
select *from (
        select *,row_number()over (partition by userid order by created_date desc) as
row_no from (select s.userid, s.product_id, s.created_date, g.gold_signup_date
from sales s inner join goldusers_signup g on s.userid=g.userid
where s.created_date<g.gold_signup_date) x) y where row_no=1;
```

OUTPUT

	userid	product_id	created_date	gold_signup_date	row_no
▶	1	2	2017-04-19	2017-09-22	1
	3	2	2016-12-20	2017-04-21	1

What is the total orders and amount spent for each member before they became a member?

```
Select s.userid, count(p.product_id) as order_purchased, sum(p.price) as
Total_amount_spent from product p inner join sales s on p.product_id = s.product_id
inner join goldusers_signup g on s.userid = g.userid  where
s.created_date<g.gold_signup_date group by userid order by userid;
```

OUTPUT

	userid	order_purchased	Total_amount_spent
▶	1	5	4030
	3	3	2720

If buying each product generates points for eg 5rs=2 zomato point and each product has different purchasing points for eg for p1 5rs=1 zomato point, for p2 10rs=5 zomato point and p3 5rs=1 zomato point.
Calculate points collected by each customers and for which product most points have been given till now ?

1.Calculated points collected by each customer .

```
Select c.userid, sum(Total_points*2.5) as Total_earnings from (
Select b.userid,b.product_name,Total_amount, floor((Total_amount/points)) as Total_points from (
select a.*, ( case when product_name='p1' then 5
                  when product_name='p2'  then 2
                  when product_name='p3' then 5 else 0 end ) as points  from (
Select s.userid, p.product_name, sum(p.price) as Total_amount from sales s inner join product p  on
s.product_id = p.product_id group by s.userid, p.product_name) a) b) c group by userid order by userid;
```

OUTPUT

	userid	Total_earnings
▶	1	4572.5
	2	1907.5
	3	4242.5

2.which product most points given till now?

```
Select * from (
select c.*, row_number() over(order by Total_points desc) as row_no  from (
select b.product_name, floor((Total_amount/points)) as Total_points  from (
select a.*, ( case when product_name='p1' then 5
                   when product_name='p2’ then 2
                   when product_name='p3' then 5 else 0 end ) as points from (
Select p.product_name, sum(p.price) as Total_amount from sales s inner join
product p on s.product_id = p.product_id group by p.product_name) a) b) c) d
                              where row_no=1;
```

OUTPUT

	product_name	Total_points	row_no
▶	p2	3045	1

In the first one year after a customer joins the fold program(including their join date)irrespective of what the customer has purchased they earn 5 zomato points for every 10rs spent who earned more and whar was their points earnings in their first year?

```
select c.userid, c.created_date, c.product_id, c.gold_signup_date, Total_points,
row_number() over ( order by Total_points desc as row_no from (
Select b.*,  floor ((Total_amount/points)) as  Total_points  from   (
select a.*,( case when product_id then 2 else 0 end) as points from (
select s.userid,s.created_date,g.gold_signup_date,s.product_id,sum(p.price) as Total_amount
from product p inner join  sales s on p.product_id = s.product_id  inner join  goldusers_signup  g
on s.userid = g.userid where created_date>=gold_signup_date and created_date<=date
Add (gold_signup_date, interval 1 year)
group by userid, created_date, gold_signup_date, product_id) a) b) c ;
```

OUTPUT

	userid	created_date	product_id	gold_signup_date	Total_points	row_no
▶	3	2017-07-12	2	2017-04-21	435	1
	1	2018-03-19	3	2017-09-22	165	2

Rank all the transaction of the customers?

```
Select *,rank() over(partition by userid order by
created_date) as rnk from sales;
```

OUTPUT

userid	created_date	product_id	rnk
1	2016-03-11	1	1
1	2016-05-20	3	2
1	2016-11-09	1	3
1	2017-03-11	2	4
1	2017-04-19	2	5
1	2018-03-19	3	6
1	2019-10-23	2	7
2	2017-08-11	2	1
2	2017-09-24	1	2
2	2018-10-09	3	3
2	2020-07-20	3	4
3	2016-10-11	1	1
3	2016-12-15	2	2
3	2016-12-20	2	3
3	2017-07-12	2	4
3	2019-12-18	1	5

Rank all the transactions for each member whenever they are a Zomato gold member -- for every non gold member transactions mark as “Na” ?

```
Select b.*, (case when gold_signup_date then rnk
 when gold_signup_date is NULL then ‘Na' end)  as rank_no from (
select a.*,rank() over(partition by userid order by created_date desc)
as rnk from (
Select s.userid, s.created_date, s.product_id, g.gold_signup_date
from sales s left join goldusers_signup g on s.userid = g.userid  and
s.created_date >= g.gold_signup_date) a) b;
```

OUTPUT

userid	created_date	product_id	gold_signup_date	rnk	rank_no
1	2019-10-23	2	2017-09-22	1	1
1	2018-03-19	3	2017-09-22	2	2
1	2017-04-19	2	NULL	3	Na
1	2017-03-11	2	NULL	4	Na
1	2016-11-09	1	NULL	5	Na
1	2016-05-20	3	NULL	6	Na
1	2016-03-11	1	NULL	7	Na
2	2020-07-20	3	NULL	1	Na
2	2018-10-09	3	NULL	2	Na
2	2017-09-24	1	NULL	3	Na
2	2017-08-11	2	NULL	4	Na
3	2019-12-18	1	2017-04-21	1	1
3	2017-07-12	2	2017-04-21	2	2
3	2016-12-20	2	NULL	3	Na
3	2016-12-15	2	NULL	4	Na
3	2016-10-11	1	NULL	5	Na


THANK YOU