# "Dinelytics: A Qustomer Intelligence for FoodTech Insights"



Dinelytics is a comprehensive SQL-based data analysis project designed to simulate customer behavior tracking and sales analytics for a modern FoodTech platform.

The project leverages transactional and membership data to uncover patterns in customer purchases, membership impact, and product popularity.

Built with a realistic schema involving users, product catalogs, purchase history, and loyalty membership (similar to Zomato Gold), Dinelytics demonstrates how data can be transformed into actionable customer intelligence.

# Project Objectives:

- Track and analyze total customer spending over time.
- Identify first-time purchases and most frequently bought products.
- Evaluate customer behavior before and after premium membership activation.
- Rank transactions and explore membership-driven sales patterns.
- Detect the most popular products overall and per user.

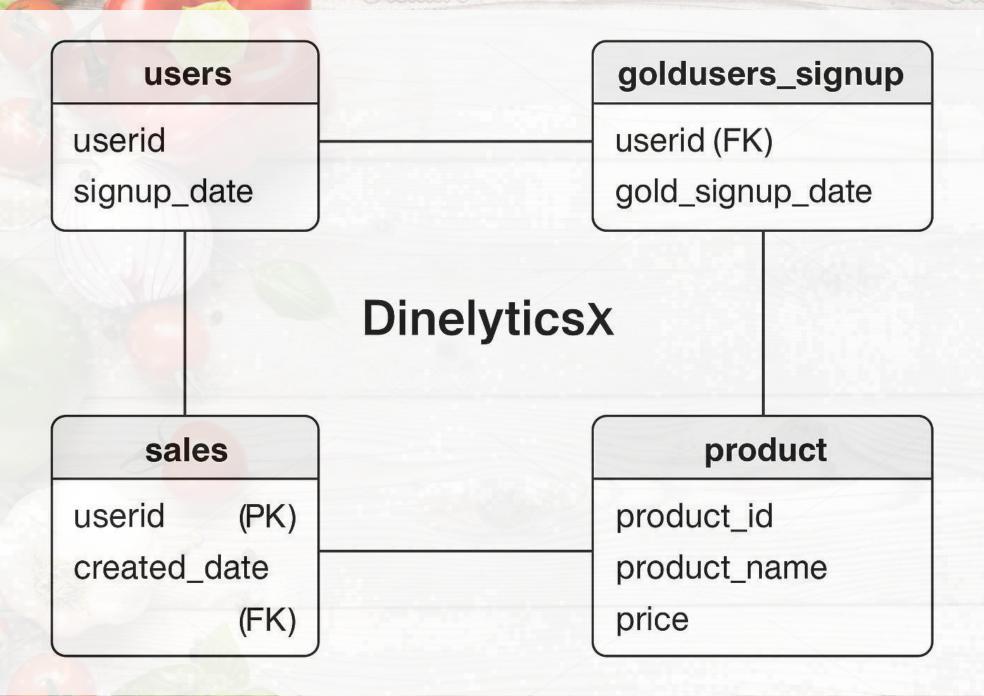


- SQL: JOINs, subqueries, window functions (RANK()), filtering, and aggregation.
- Relational Data Modeling: Connecting users, sales, products, and gold memberships.
- Customer Analytics: Lifecycle tracking, membership impact analysis.
- Business Logic Implementation: Realworld scenarios like upselling, loyalty effect, and purchase trends

# Business Relevance:

Dinelytics showcases how companies in the FoodTech industry can harness raw transactional data to understand user behavior, personalize experiences, and refine loyalty programs. The insights extracted from this project can directly support strategies in product optimization, targeted marketing, and customer retention.

# Entity Relationship Diagram (ERD)



# Drop and Create Table

 userid
 gold\_signup\_date

 1
 1
 2017-09-22

 2
 3
 2017-04-21

Gold user signup table

```
drop table if exists users;
CREATE TABLE users(userid integer, signup_date date);
INSERT INTO users(userid, signup_date)
VALUES (1,'09-02-2014'),
(2,'01-15-2015'),
(3,'04-11-2014');
1
```

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

User table

# Drop and Create Table

```
drop table if exists sales;
CREATE TABLE sales(userid integer, created_date date, product_id integer);
```

```
INSERT INTO sales(userid, created_date, product_id)
VALUES (1, '04-19-2017', 2),
(3, '12-18-2019', 1),
(2,'07-20-2020',3),
(1, '10-23-2019', 2),
(1, '03-19-2018', 3),
(3, '12-20-2016', 2),
(1, '11-09-2016', 1),
(1,'05-20-2016',3),
(2, '09-24-2017', 1),
(1, '03-11-2017', 2),
(1, '03-11-2016', 1),
(3, '11-10-2016', 1),
(3, '12-07-2017', 2),
(3, '12-15-2016', 2),
(2, '11-08-2017', 2),
(2, '09-10-2018', 3);
```

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

### Result

	userid	created_date	product_id
1	1	2017-04-19	2
2	3	2019-12-18	1
3	2	2020-07-20	3
4	1	2019-10-23	2
5	1	2018-03-19	3
6	3	2016-12-20	2
7	1	2016-11-09	1
8	1	2016-05-20	3
9	2	2017-09-24	1
10	1	2017-03-11	2
11	1	2016-03-11	1
12	3	2016-11-10	1
13	3	2017-12-07	2
14	3	2016-12-15	2
15	2	2017-11-08	2
16	2	2018-09-10	3

Sales table

```
Product table

drop table if exists product;

CREATE TABLE product(product_id integer, product_name text, price integer)

INSERT INTO product(product_id, product_name, price)

VALUES

(1,'p1',980),
(2,'p2',870),
```

(3, 'p3', 330);

--1. What is the total amount each customer spent on ordering from these food tech company?

Select a.userid, SUM(b.price) Total\_amt\_spent from sales a inner join product b on a.product\_id=b.product\_id group by a.userid

### Result

	userid	Total_amt_spent
1	1	5230
2	2	2510
3	3	4570

-- 2. How many days has each customer visited on a food tech platform?

Select userid, count(distinct created\_date) Distinct\_Days\_visited from sales
group by userid

Result

	userid	Distinct_Days_visited
1	1	7
2	2	4
3	3	5

--3. What was the first product purchased by each customer?

```
Select * from
    ( select * , RANK () over (partition by userid order by created_date)rnk
    from sales)a
where rnk=1
Result
```

	userid	created_date	product_id	rnk
1	1	2016-03-11	1	1
2	2	2017-09-24	1	1
3	3	2016-11-10	1	1

--4. What is the most purchased item on the menu and how many times was it purchased by all customers?

```
Select userid, count (product_id) cnt
from sales
where product_id=(select top 1 product_id from sales group by product_id
order by count(product_id) desc) group by userid
```

	userid	cnt
1	1	3
2	2	1
3	3	3

--5. Which item was the most popular in each customer?

--6. Which item was purchased first by the customer after they became a member?

```
Select * from(
    select c. * , RANK () over (partition by userid order by created_date) rnk
    from(
        select a.userid, a.created_date, a.product_id, b.gold_signup_date
        from sales a inner join goldusers_signup b
        on a.userid=b.userid and created_date >= gold_signup_date)c)d
where rnk=1
```

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

--7. Which item purchased just before the customer became a member?

```
Select * from(
    select c. * , RANK () over (partition by userid order by created_date desc) rnk
    from(
        select a.userid, a.created_date, a.product_id, b.gold_signup_date
        from sales a inner join goldusers_signup b
        on a.userid=b.userid and created_date <= gold_signup_date)c)d
where rnk=1</pre>
```

### Result

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

--8. What is the total orders and amount spent for each member before they became the member?

```
Select userid , COUNT(created_date) Order_purchased , SUM(price) Total_amt_spent
from(
    select c.* , d.price
    from(
        select a.userid, a.created_date, a.product_id, b.gold_signup_date
        from sales a inner join goldusers_signup b
        on a.userid=b.userid and created_date <= gold_signup_date)c
        inner join product d on c.product_id=d.product_id) e
group by userid</pre>
```

	userid	Order_purchased	Total_amt_spent
1	1	5	4030
2	3	3	2720

-- 9. Rank all the transaction of the customers?

```
Select * ,
RANK() over(partition by userid order by created_date) rnk
from sales
```

	userid	created_date	product_id	rnk
1	1	2016-03-11	1	1
2	1	2016-05-20	3	2
3	1	2016-11-09	1	3
4	1	2017-03-11	2	4
5	1	2017-04-19	2	5
6	1	2018-03-19	3	6
7	1	2019-10-23	2	7
8	2	2017-09-24	1	1
9	2	2017-11-08	2	2
10	2	2018-09-10	3	3
11	2	2020-07-20	3	4
12	3	2016-11-10	1	1
13	3	2016-12-15	2	2
14	3	2016-12-20	2	3
15	3	2017-12-07	2	4
16	3	2019-12-18	1	5

--10. Rank all the transaction for each customer whenever they took a premium membership of a foodtech company and for every non member transaction mark as 'na'?

```
Select e.*,
case when rnk=0 then 'na' else rnk end as rnkk
from(
    select c.*, CAST (( case when gold_signup_date is null then 0
    else rank() over (partition by userid order by created_date desc) end ) as varchar)as rnk
    from(
        select a.userid, a.created_date, a.product_id, b.gold_signup_date
        from sales a left join goldusers_signup b
        on a.userid=b.userid and created_date >= gold_signup_date)c)e
```

	userid	created_date	product_id	gold_signup_date	rnk	rnkk
1	1	2019-10-23	2	2017-09-22	1	1
2	1	2018-03-19	3	2017-09-22	2	2
3	1	2017-04-19	2	NULL	0	na
4	1	2017-03-11	2	NULL	0	na
5	1	2016-11-09	1	NULL	0	na
6	1	2016-05-20	3	NULL	0	na
7	1	2016-03-11	1	NULL	0	na
8	2	2020-07-20	3	NULL	0	na
9	2	2018-09-10	3	NULL	0	na
10	2	2017-11-08	2	NULL	0	na
11	2	2017-09-24	1	NULL	0	na
12	3	2019-12-18	1	2017-04-21	1	1
13	3	2017-12-07	2	2017-04-21	2	2
14	3	2016-12-20	2	NULL	0	na
15	3	2016-12-15	2	NULL	0	na
16	3	2016-11-10	1	NULL	0	na



The Dinelytics project successfully simulates a realworld FoodTech environment to uncover deep customer insights using SQL. Through structured analysis of user transactions, gold membership patterns, and product sales, the project demonstrates how raw data can drive meaningful business decisions. The queries implemented reveal behavioral trends, membership impact, and product engagement metrics — essential for optimizing loyalty programs and product strategy.

This project highlights the practical application of SQL in solving real-life business problems by transforming relational data into actionable insights.

