Case Study : Ecommerce Order Analysis

This repository contains an analysis of ecommerce order data to answer specific questions.

The dataset consists of two tables: Order1 and Order2. Below is a brief description of the schema.

- **Order1**
  - OrderId
  - OrderItemId
  - QuantityOrdered
  - ItemPrice
  - PromotionDiscount
  - batch_id

- **Order2**
  - OrderId
  - Purchasedate
  - OrderStatus
  - batch_id

Questions to Answer

1. Calculate the Total order quantity, GMV, and Net sales.
2. Calculate Total order quantity, GMV, and Net sales for the month of October 2022 where the order status is not equal to "Canceled."

Que-1: Calculate the Total order quantity, GMV and Net sales, where GMV is the summation of all the order values, and Net sale is the values after the discount.

Approach -
1. Assumed flat discount value per order per item_sale (item_quantity * price)
2. Assumed different batch number for each item per order

Query -

```sql
with t1 as
    (select
    OrderId,
    OrderItemId,
    batch_id, QuantityOrdered,
    QuantityOrdered * ItemPrice as ordervalue,
    (QuantityOrdered * ItemPrice) - Cast(json_value(PromotionDiscount, '$.Amount') as float64) as
net_sale
    from statfinity_sql_case.order_1),

t2 as
    (select
    OrderId,
    OrderItemId,
    max(batch_id) as batch_id,
    sum(QuantityOrdered) as QuantityOrdered,
    sum(ordervalue) as ordervalue,
    sum(net_sale) as net_sale
    from t1
    group by 1, 2)

select
  sum(QuantityOrdered) as TotalOrderQuantity,
  round(sum(ordervalue), 2) as GMV,
  round(sum(net_sale), 2) as TotalSales
from t2;
```

Q2. Calculate Total order quantity, GMV and Net sales for the month of October 2022 where the order status is not equal to "Canceled".

Query -
```sql
with t1 as (
  select
    o1.OrderId, o1.OrderItemId,
    o1.batch_id, o1.QuantityOrdered,
    o1.QuantityOrdered * o1.ItemPrice as ordervalue,
    (o1.QuantityOrdered * o1.ItemPrice) - cast(json_value(
    o1.PromotionDiscount, '$.Amount') as float64) AS net_sale,
    o2.Purchasedate,
    o2.OrderStatus
  from statfinity_sql_case.order_1 as o1
  join statfinity_sql_case.order_2 as o2
  on o1.OrderId = o2.OrderId),

t2 as
  (select
    OrderId,OrderItemId,
    max(batch_id) as batch_id,
    sum(QuantityOrdered) as QuantityOrdered,
    sum(ordervalue) as ordervalue,
    sum(net_sale) as net_sale
  from t1
  where extract(month from Purchasedate) = 10 and
    extract(year from Purchasedate) = 2022 and
    OrderStatus != 'Canceled'
  group by 1,2)

select
 sum(QuantityOrdered) AS TotalOrderQuantity,
 round(sum(ordervalue), 2) AS GMV,
 round(sum(net_sale), 2) AS TotalSales
from t2;
```