E-Commers Analysis

The dataset comprises nine tables related to Olist, an e-commerce platform. The `olist_customers_dataset` contains customer details such as ID, unique ID, and location. The `olist_geolocation_dataset` provides geospatial information about customers. The `olist_order_items_dataset` details individual items within orders, including product and seller information. The `olist_order_reviews_dataset` contains reviews with scores and comments for orders. The `olist_orders_dataset` table encompasses order-related information like status and timestamps. The `olist_products_dataset` includes details about products, such as category and dimensions. The `olist_sellers_dataset` offers information about sellers, including location. The `product_category_name_translation` table translates product categories. Lastly, the `olist_order_payments_dataset` provides payment details for orders.

Analysts can utilize these tables for various analyses. For example, they can assess customer distribution across cities, identify top-selling products or sellers, analyze review scores, and evaluate order fulfillment timelines. Additionally, insights can be drawn regarding payment types, revenue per seller, and product category performance. By merging and querying these tables, analysts can gain a comprehensive understanding of Olist's operations, enabling data-driven decision-making to enhance customer satisfaction, optimize logistics, and improve overall business efficiency. The dataset provides a rich source for exploring patterns, trends, and performance metrics within the e-commerce ecosystem.

```sql
select * from olist_customers_dataset;
select * from olist_geolocation_dataset;
select * from olist_order_items_dataset;
select * from olist_order_payments_dataset;
select * from olist_order_reviews_dataset;
select * from olist_orders_dataset;
select * from olist_products_dataset;
select * from olist_sellers_dataset;
select * from product_category_name;
```

```sql
--1.Retrieve the total number of unique customers in the
olist_customers_dataset.--

select
    count (distinct(customer_id)) as unique_customers
from olist_customers_dataset;


--2.Find the top 5 cities with the highest number of customers.--

select top 5
         COUNT(customer_id) as Customer_count,
         customer_city from olist_customers_dataset
       group by
          customer_city
       order by
          Customer_count desc;


--3.Calculate the average freight value for each product category in the
olist_order_items_dataset.--

select
     p.product_category_name,
       AVG(oi.freight_value) as avg_freight
from olist_order_items_dataset oi
     inner join olist_products_dataset p on oi.product_id = p.product_id
     group by
       p.product_category_name
     order by
       avg_freight desc;


--4.Identify the sellers who have completed the most orders. Display the
top 3 sellers.--

select top 3
        count(distinct oi.order_id) as order_count,
            oi.seller_id from olist_order_items_dataset oi
inner join olist_sellers_dataset os on oi.seller_id=os.seller_id
        group by
            oi.seller_id
        order by
            order_count desc;


--5.Determine the average product weight for each product category in the
olist_products_dataset.--

select
     product_category_name,
     AVG(product_weight_g) as Avg_Product_wt
from olist_products_dataset
     group by
```

```
        product_category_name
      order by
        Avg_Product_wt desc;


--6.List the top 10 products with the highest average review scores.--

select top 10
            AVG(ord.review_score) as Avg_review_score,
                oi.product_id
from olist_order_reviews_dataset ord
inner join olist_order_items_dataset oi on ord.order_id=oi.order_id
            group by
                oi.product_id,
                ord.order_id
            order by
                Avg_review_score desc


--7.Find the percentage of orders delivered on time
(delivered_customer_date within estimated_delivery_date)
--for each seller. Display the top 5 sellers with the highest percentage.-
-


--8.Identify the most common payment type used by customers for orders.--

select count(order_id) as order_count, payment_type from
olist_order_payments_dataset
group by payment_type
order by order_count desc;


--9.Calculate the total revenue generated by each seller. Display the top
5 sellers with the highest revenue.—

select * from olist_order_items_dataset;

select top 5 seller_id, SUM(price+freight_value) as Total_revenue from
olist_order_items_dataset
group by seller_id
order by Total_revenue desc


--10.Find the average number of products per order in the
olist_order_items_dataset.—


select (Sum(order_item_id)/Count(order_id)) as avg_orders,order_id from
olist_order_items_dataset
group by order_id
order by avg_orders desc;
```

```sql
--11.Retrieve the order details with the longest shipping times
--(shipping_limit_date to delivered_customer_date) for each product
category.—


select order_id, order_approved_at, order_delivered_customer_date,
DATEDIFF(DAY,order_approved_at,order_delivered_customer_date) as
Shipping_time
from olist_orders_dataset
order by Shipping_time desc;


--12.Calculate the average review score for orders with multiple items
(more than one order_item_id).--

select ord.order_id,Avg(ord.review_score) as Avg_review,oi.order_item_id
from olist_order_reviews_dataset ord inner join
olist_order_items_dataset oi on (ord.order_id=oi.order_id)
group by ord.order_id,ord.review_score,oi.order_item_id
having oi.order_item_id > 1
order by Avg_review desc



--13.Identify the customers who have made the highest total payment value.
Display the top 3 customers.--

select top 3
         op.order_id,
             op.payment_value as high_payment_value,
             ords.customer_id
from olist_order_payments_dataset op
         inner join olist_orders_dataset ords on
(op.order_id=ords.order_id)
         group by
             ords.customer_id,
             op.order_id,
             payment_value
         order by
             high_payment_value desc;


--14.Find the sellers who have products in the most diverse product
categories. Display the top 3 sellers.--

select
     COUNT(pro.product_id) as product_count,
       pro.product_category_name,
       oi.seller_id
from olist_products_dataset pro
     inner join olist_order_items_dataset oi on
       (pro.product_id=oi.product_id)
     group by
       pro.product_id,
```

```
        pro.product_category_name,
        oi.seller_id
      order by
        product_count desc


--15.Calculate the average payment value for each payment type in the
olist_order_payments_dataset.—


select
      payment_type,
        AVG(payment_value) as avg_payment
from olist_order_payments_dataset
      group by
        payment_type
      order by
        avg_payment desc


--16.Determine the average time taken to approve an order
--(order_approved_at - order_purchase_timestamp) for each seller.—


select
      order_id,order_purchase_timestamp,
        order_approved_at,
      avg(datediff(MINUTE,order_purchase_timestamp,order_approved_at)) as
timediffs
from olist_orders_dataset
   group by
   order_id,order_purchase_timestamp,
   order_approved_at
   order by timediffs desc;
```