

FROM DATA TO DECISIONS: OLIST'S E-COMMERCE PLATFORM

PERFORMANCE ANALYSIS

Project Objective: E-commerce Performance Analysis

Objective Overview: As the e-commerce landscape becomes increasingly competitive, understanding key performance metrics is essential for driving growth and maximizing profitability. The objective of this analysis is to provide a comprehensive overview of the client's e-commerce performance by leveraging top-line Key Performance Indicators (KPIs). These insights will empower the client to make data-driven decisions to enhance customer acquisition, retention, and overall sales performance.

Key Performance Indicators (KPIs) Covered:

1. **Total Sales Revenue:** Measure the overall income generated from product sales to assess business growth and market demand.
2. **Number of Orders:** Track the volume of transactions to understand customer purchasing behaviour and sales trends.
3. **Average Order Value (AOV):** Calculate the average amount spent per transaction to identify opportunities for upselling and cross-selling.
4. **Product Sales Performance:** Track product sales by order volume and revenue generated for top-performing products for each year, quarter and month.
5. **Active Sellers:** Analyze the number of sellers/vendors who use the platform more than once for the period of the analysis.
6. **Payment Type Usage:** Identify the most used or preferred payment type by customers for transactions.
7. **Profit Margin %:** Measure the profitability of products concerning sales on the platform.
8. **Customer Retention Rate:** Measure the percentage of repeat customers to understand and improve customer loyalty.

Project Goals:

1. Provide a clear and actionable overview of e-commerce performance.
2. Identify strengths and areas for improvement in customer retention, and sales strategies.
3. Enhance customer experience and satisfaction through data-driven insights.
4. Support strategic decision-making with detailed and accurate performance metrics.

Data Analysis

E-commerce Performance Analysis using SQL

For this analysis, I will be querying the dataset to derive insights on sales made on the platform for the recorded period.

The following questions will be used to derive these insights. A total of 13 questions will be asked in this analysis. They are as follows:

1. WHAT IS THE TOTAL REVENUE GENERATED BY OLIST, AND HOW HAS IT CHANGED OVER TIME?
2. HOW MANY ORDERS WERE PLACED ON OLIST, AND HOW DOES THIS VARY BY MONTH?
3. WHAT ARE THE MOST POPULAR PRODUCT CATEGORIES ON OLIST? HOW DO THEIR SALES VOLUMES COMPARE TO EACH OTHER?
4. WHAT IS THE AVERAGE ORDER VALUE (AOV) ON OLIST? HOW DOES THIS VARY BY PRODUCT CATEGORY OR PAYMENT METHOD?
5. HOW MANY SELLERS ARE ACTIVE ON OLIST, AND HOW DOES THIS NUMBER CHANGE OVER TIME?
6. WHAT IS THE DISTRIBUTION OF SELLER RATINGS ON OLIST, AND HOW DOES THIS IMPACT SALES PERFORMANCE?
7. HOW MANY CUSTOMERS HAVE MADE REPEAT PURCHASES ON OLIST, AND WHAT PERCENTAGE OF TOTAL SALES DO THEY ACCOUNT FOR?
8. WHAT IS THE AVERAGE CUSTOMER RATING FOR PRODUCTS SOLD ON OLIST, AND HOW DOES THIS IMPACT SALES PERFORMANCE?
9. WHAT IS THE AVERAGE ORDER CANCELLATION RATE ON OLIST, AND HOW DOES THIS IMPACT SELLER PERFORMANCE?
10. WHAT ARE THE TOP-SELLING PRODUCTS ON OLIST, AND HOW HAVE THEIR SALES TRENDS CHANGED OVER TIME?
11. WHICH PAYMENT METHODS ARE MOST COMMONLY USED BY OLIST CUSTOMERS, AND HOW DOES THIS VARY ACROSS PRODUCT CATEGORY OR GEOGRAPHIC REGION?
12. WHICH PRODUCT CATEGORIES HAVE THE HIGHEST PROFIT MARGINS ON OLIST?
13. WHICH GEOLOCATION HAS HIGH CUSTOMER DENSITY? CALCULATE CUSTOMER RETENTION RATE ACCORDING TO GEOLOCATIONS.

Below are the queries used to derive the results from the questions asked to get the required insights.

The SQL script for the data analysis on the dataset can be found by clicking on this [link](#).

1. WHAT IS THE TOTAL REVENUE GENERATED BY OLIST, AND HOW HAS IT CHANGED OVER TIME?

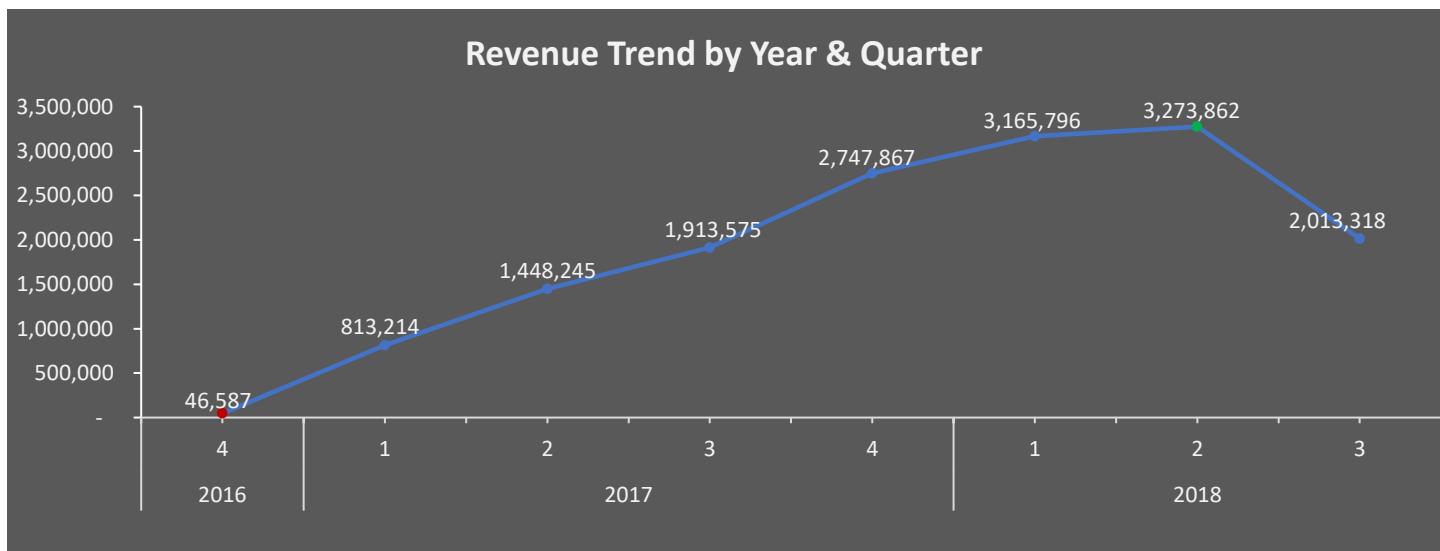
Solution:

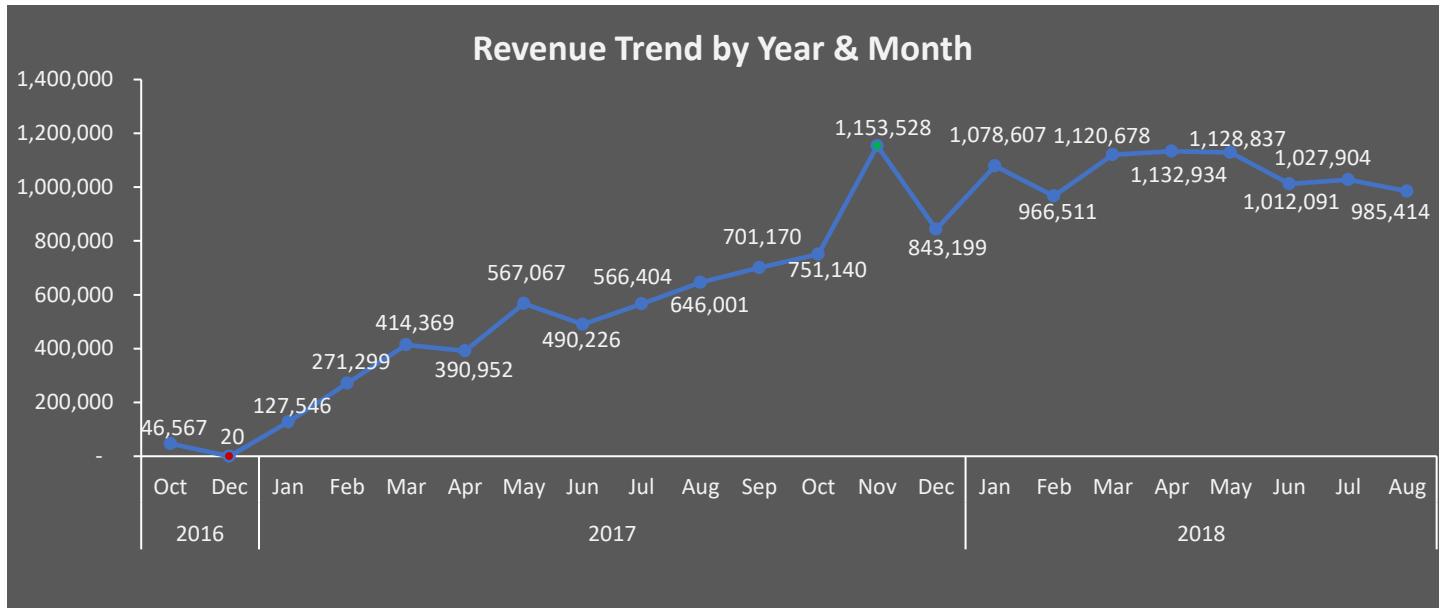
```
Total Revenue  
SELECT  
    ROUND(SUM(opa.payment_value), 0) AS total_revenue  
FROM  
    olist_orders AS oo  
    INNER JOIN  
    olist_payments AS opa ON oo.order_id = opa.order_id  
WHERE  
    oo.order_status = 'delivered'  
    AND order_delivered_customer_date IS NOT NULL;
```

Output: Total revenue is R\$15,422,462.

Total Revenue
R\$15,422,462.00

Total Revenue Trend (Year, Month, Quarter)





2. HOW MANY ORDERS WERE PLACED ON OLIST, AND HOW DOES THIS VARY BY MONTH OR SEASON?

Solution: For this question, 'Canceled' orders were omitted

Output:

A total of **98,816** orders were placed on Olist.

Total Orders
98,816

```
... Total Orders

SELECT
    COUNT(*) AS total_orders
FROM
    olist_orders
WHERE
    olist_orders.order_status <> 'canceled'
    AND
    olist_orders.order_delivered_customer_date IS NOT
NULL;
```

Total orders trend (Year, Quarter, Month)



QUESTIONS 3 AND 10 BOTH SEEK TO ANSWER THE SAME QUESTION.

THE SECOND PART OF QUESTION 10 WHICH LOOKS AT THE SALES PERFORMANCE OF PRODUCTS WILL BE ANSWERED SEPARATELY.

Q3. WHICH ARE THE MOST POPULAR PRODUCT CATEGORIES ON OLIST?

Q10. WHICH IS THE TOP-SELLING PRODUCTS ON OLIST, AND HOW HAVE THEIR SALES TREND CHANGED OVER TIME?

Solution: First we will look at the total number of product orders

```
Total Number of Products

SELECT
    COUNT(oi.product_id) AS total_num_products
FROM
    olist_items AS oi
    JOIN
    olist_orders AS oo ON oi.order_id = oo.order_id
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL;
```

Output: The total number of products ordered on Olist is **112,108**.

Total Number of Products
112,108

For the most popular products/top-selling products

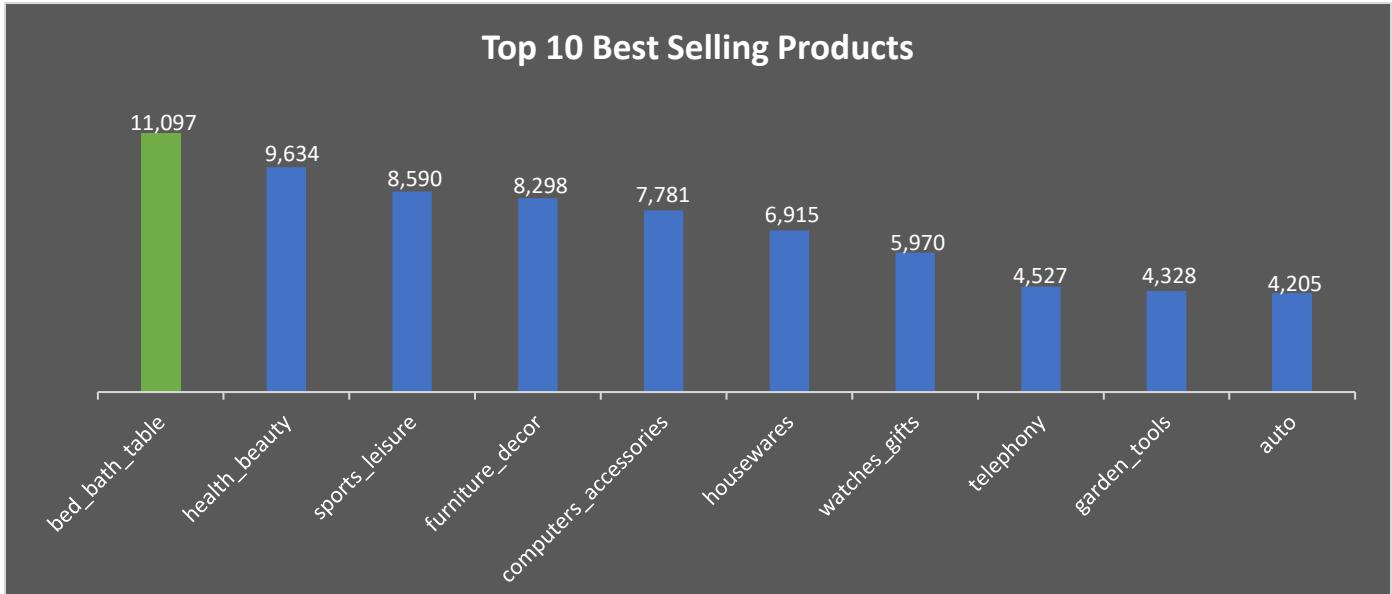
Solution:

```
Popular / Top Selling Products

WITH total_orders AS (
    SELECT
        COUNT(oo.order_id) AS total_num_orders
    FROM
        olist_orders AS oo
    JOIN olist_items AS oi ON oo.order_id = oi.order_id
    JOIN olist_products AS op ON oi.product_id = op.product_id
    WHERE
        oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
)
SELECT
    op.product_category_name_english AS product_name,
    COUNT(oo.order_id) AS num_of_orders,
    ROUND((COUNT(oo.order_id) / total_orders.total_num_orders) * 100, 2) AS percentage
FROM
    olist_orders AS oo
    JOIN
    olist_items AS oi ON oo.order_id = oi.order_id
    JOIN
    olist_products AS op ON oi.product_id = op.product_id
    CROSS JOIN
    total_orders
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
    product_name,
    total_num_orders
ORDER BY percentage DESC;
```

Output:

The most popular/top-selling product categories on Olist are bed_bath_table, health_beauty and sports_leisure. This concerns the number of orders placed.



For the best performing / top selling products for each year

Solution:

```
Top Selling / Performing Products by Year & Rank

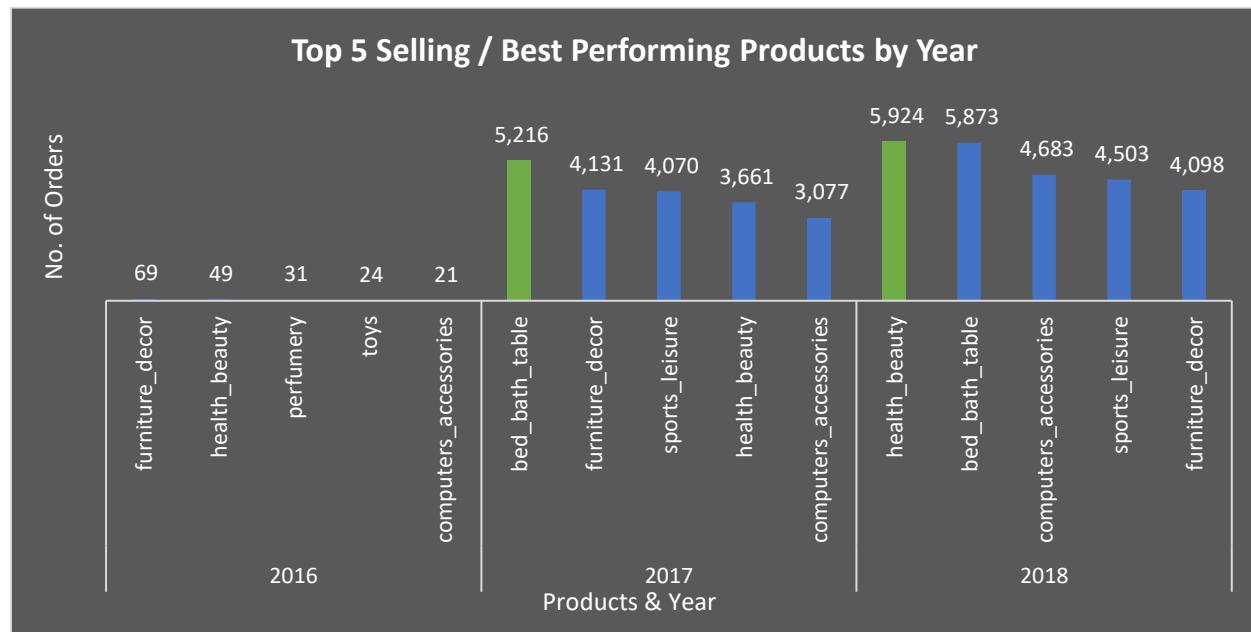
SELECT
    YEAR(oo.order_purchase_timestamp) AS order_year,
    op.product_category_name_english AS product_name,
    COUNT(oo.order_id) AS num_of_orders,
    RANK() OVER(PARTITION BY YEAR(oo.order_purchase_timestamp) ORDER BY COUNT(oo.order_id) DESC)
    AS ranking
FROM
    olist_orders AS oo
JOIN
    olist_items AS oi ON oo.order_id = oi.order_id
JOIN
    (SELECT
        product_id,
        product_category_name_english
        FROM olist_products) AS op ON oi.product_id = op.product_id
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
    order_year, product_name
ORDER BY
    order_year, ranking;
```

Output:

In **2016**, the best-performing products were furniture_decor, health_beauty, perfumery, toys and computers_accessories.

In **2017**, the best-performing products were bed_bath_table, furniture_decor, sports_leisure, health_beauty and computers_accessories.

In **2018**, the best-performing products were health_beauty, bed_bath_table, computers_accessories, sports_leisure and furniture_decor.



4. WHAT IS THE AVERAGE ORDER VALUE (AOV)? HOW DOES THIS VARY BY PRODUCT CATEGORY OR PAYMENT METHOD?

Solution:

AOV is a crucial metric in understanding the overall performance of products compared to the cost per order (CPO).

It allows a business to determine the performance of each product.

AOV = total revenue / total orders

CPO = total cost of product / total orders

Output:

The average order value (AOV) is **R\$160.28**, the cost per order (CPO) is also **R\$160.25** and the **Profit_Per_Order** is **R\$0.03**

AOV	CPO	PROFIT PER ORDER
R\$160.28	R\$160.25	R\$0.03

```

WITH payment_totals AS(
  select
    oo.order_id,
    sum(opa.payment_value) as total_payment
  from
    olist_orders as oo
    join olist_payments as opa on oo.order_id = opa.order_id
  where
    oo.order_status <> 'canceled'
    and oo.order_delivered_customer_date is not null
  GROUP BY
    oo.order_id
),
cost_totals as (
  select
    oo.order_id,
    sum(oi.price + oi.freight_value) as total_cost
  from
    olist_orders as oo
    join olist_items as oi on oo.order_id = oi.order_id
  where
    oo.order_status <> 'canceled'
    and oo.order_delivered_customer_date is not null
  GROUP BY
    oo.order_id
)
select
  round(sum(total_payment) / count(oo.order_id), 2) as AOV,
  round(sum(total_cost) / count(oo.order_id), 2) as CPO,
  round(sum(total_payment) / count(oo.order_id) - sum(total_cost) / count(oo.order_id), 2) as
  profit_per_order
from
  olist_orders as oo
join
  payment_totals as pt on oo.order_id = pt.order_id
join
  cost_totals as ct on oo.order_id = ct.order_id;

```

HOW DOES THE AVERAGE ORDER VALUE (AOV) VARY BY PRODUCT CATEGORY?

Solution:

```

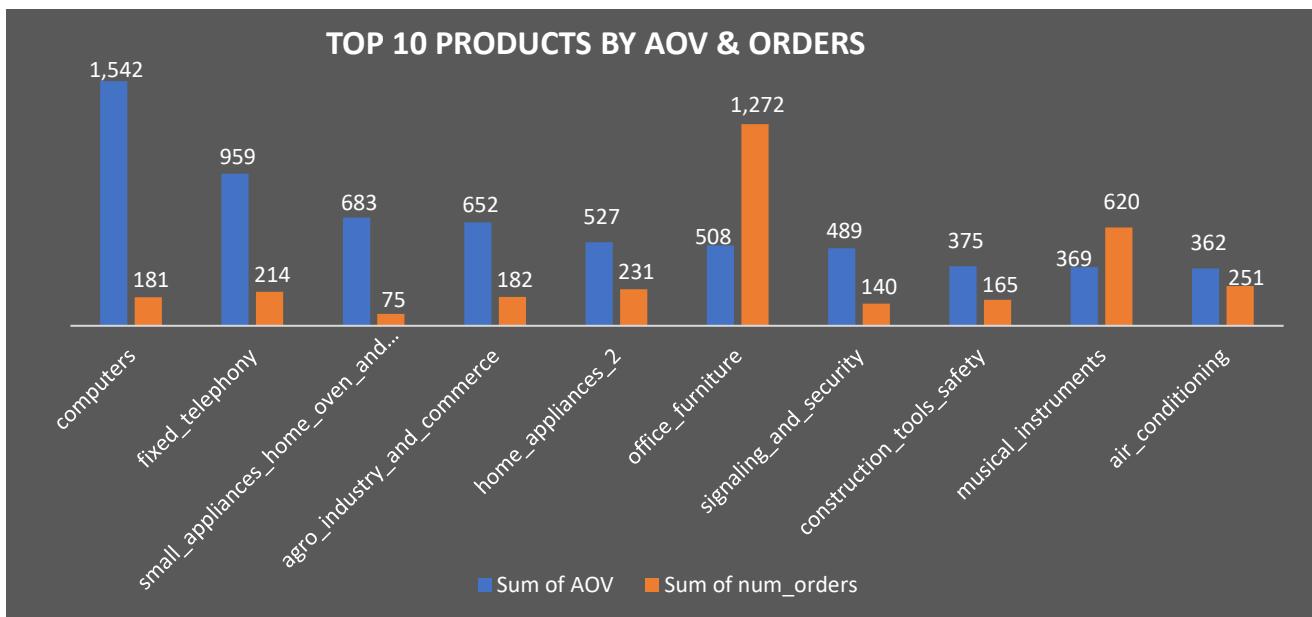
AOV by Product

SELECT
  op.product_category_name_english AS product_name,
  ROUND(SUM(opa.payment_value) / COUNT(DISTINCT oo.order_id), 0) AS AOV,
  COUNT(DISTINCT oo.order_id) AS num_orders
FROM
  olist_orders AS oo
    JOIN olist_items AS oi ON oo.order_id = oi.order_id
    JOIN olist_products AS op ON oi.product_id = op.product_id
    JOIN olist_payments AS opa ON oo.order_id = opa.order_id
WHERE
  oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
  op.product_category_name_english
ORDER BY
  AOV DESC;

```

Output:

The product category with the highest AOV is **computers** with an AOV of **1,542**.



HOW DOES THE AVERAGE ORDER VALUE (AOV) VARY BY PAYMENT TYPE?

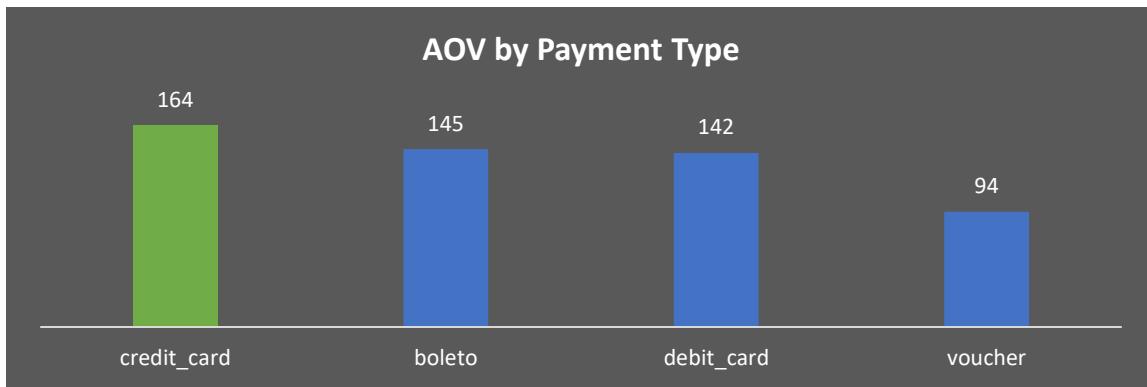
Solution:

```
AOV by Payment Type

SELECT
    opa.payment_type,
    ROUND(SUM(opa.payment_value) / COUNT(DISTINCT oo.order_id),
        0) AS AOV
FROM
    olist_orders AS oo
    JOIN
    olist_payments AS opa ON oo.order_id = opa.order_id
WHERE
    oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY opa.payment_type
ORDER BY AOV DESC;
```

Output:

credit_card payments had the highest average order value with **R\$164**, followed by boleto with an average order value of **R\$145**, debit_card with **R\$142** and then voucher payments with an AOV of **R\$94**.



5. HOW MANY SELLERS ARE ACTIVE ON OLIST, AND HOW DOES THIS NUMBER CHANGE OVER TIME?

Solution:

```

Active Sellers

SELECT COUNT(seller_id) as num_active_sellers
FROM (
    SELECT seller_id,
        (DATEDIFF(MAX(order_purchase_timestamp), MAX(previous_order_date))) as days_between_orders
    FROM(
        SELECT oi.seller_id, oo.order_id, oo.order_purchase_timestamp,
            LAG(oo.order_purchase_timestamp, 1)
            OVER(PARTITION BY oi.seller_id ORDER BY oo.order_purchase_timestamp) AS
        previous_order_date
        FROM olist_orders AS oo
        JOIN olist_items AS oi ON oi.order_id = oo.order_id
        JOIN olist_sellers AS os ON oi.seller_id = os.seller_id
        WHERE oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
        ORDER BY oi.seller_id, oo.order_purchase_timestamp desc
    ) AS pre_order_date
    GROUP BY seller_id
    HAVING
        DATEDIFF(Max(order_purchase_timestamp), Max(previous_order_date)) <=30
        AND
        DATEDIFF(MAX(order_purchase_timestamp), MAX(previous_order_date)) IS NOT NULL
        ORDER BY
        MAX(order_purchase_timestamp), MAX(previous_order_date)
    ) AS active_seller;

```

Output:

The number of active sellers on Olist is **1948** for the period this data was captured.

Total Number of Active Sellers
1,948

HOW HAS THE NUMBER OF ACTIVE SELLERS ON OLIST CHANGED OVER TIME?

Solution:

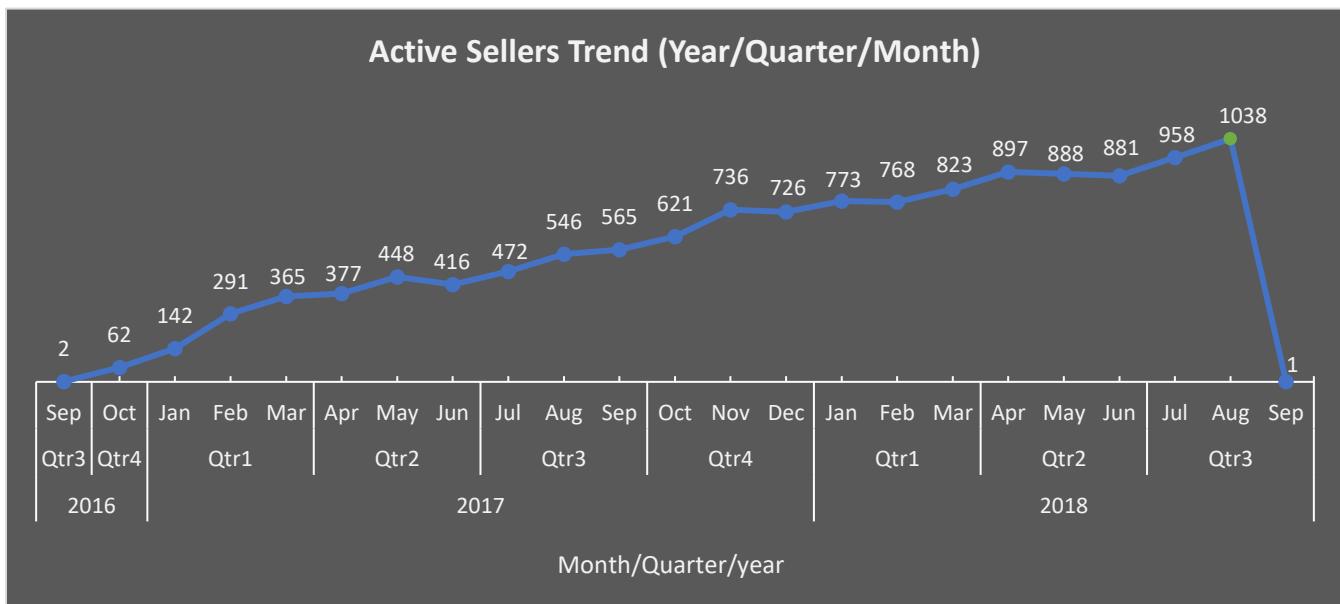
WE WILL LOOK AT THIS TREND CONCERNING YEAR, QUARTER AND MONTH

```
Active Sellers Trend

SELECT
    the_year,
    the_quarter,
    the_month,
    COUNT(DISTINCT seller_id) as active_seller_count
FROM (
    SELECT
        seller_id,
        YEAR(order_purchase_timestamp) as the_year,
        QUARTER(order_purchase_timestamp) as the_quarter,
        MONTH(order_purchase_timestamp) as the_month
    FROM (
        SELECT
            oi.seller_id,
            oo.order_purchase_timestamp,
            LAG(oo.order_purchase_timestamp, 1) OVER (PARTITION BY oi.seller_id ORDER BY
            oo.order_purchase_timestamp) AS previous_order_date
        FROM
            olist_orders as oo
        JOIN
            olist_items as oi ON oi.order_id = oo.order_id
        JOIN
            olist_sellers as os ON oi.seller_id = os.seller_id
        WHERE
            oo.order_status <> 'canceled'
            AND oo.order_delivered_customer_date IS NOT NULL
    ) as pre_order_date
    GROUP BY
        seller_id,
        the_year,
        the_quarter,
        the_month
    HAVING
        DATEDIFF(MAX(order_purchase_timestamp), MAX(previous_order_date)) <= 30
        AND MAX(order_purchase_timestamp) IS NOT NULL
) as sub
GROUP BY
    the_year, the_quarter, the_month
ORDER BY
    the_year, the_quarter, the_month;
```

Output:

The month of August in the third quarter of **2018** recorded the highest number of active sellers with a total number of **1038**.



6. WHAT IS THE DISTRIBUTION OF SELLER RATINGS ON OLIST, AND HOW DOES THIS IMPACT SALES PERFORMANCE?

Solution:

```

review score by numbers & percentage

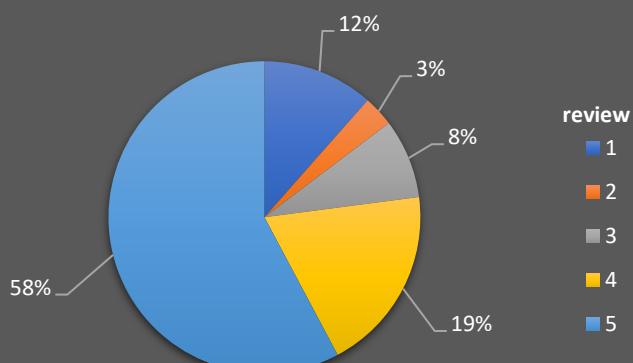
SELECT
    review_score,
    COUNT(*) AS num_of_reviews,
    ROUND((COUNT(*) / (SELECT
        COUNT(*)
    FROM
        olist_reviews) * 100),
    2) AS review_percentage
FROM
    olist_reviews
GROUP BY review_score
ORDER BY review_score DESC;

```

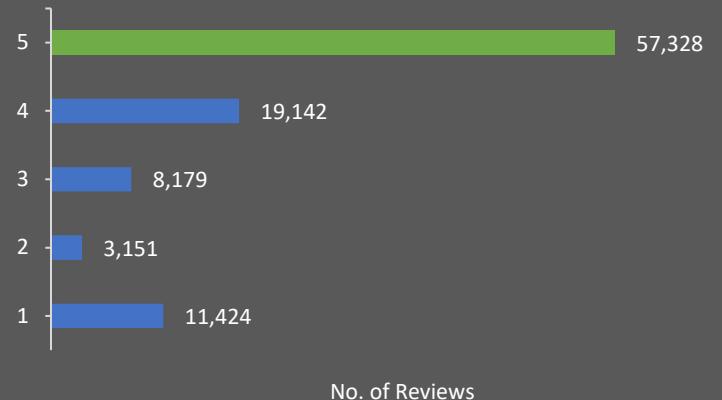
Output:

57.78% of customers were highly satisfied with the online shopping platform giving a review rating of 5. 11.51% gave the platform a rating of 1.

Review Score Distribution



Review score by Count



HOW HAS THE IMPACT OF THESE RATINGS BEEN ON SALES PERFORMANCE ON THE OLIST PLATFORM?

Solution:

To join the correct rows in the tables, views for review and payment tables were created and joined

Review view table:

```
Reviw view
CREATE OR REPLACE VIEW review AS
SELECT
    order_id, ROUND(AVG(review_score), 0) AS
review_score
FROM
    olist_reviews
WHERE
    order_id IN (SELECT
        order_id
    FROM
        olist_orders
    WHERE
        order_status <> 'canceled'
        AND
        order_delivered_customer_date IS NOT NULL)
    GROUP BY order_id;
```

Payment view table:

```
Payment view
CREATE OR REPLACE VIEW payment AS
SELECT
    order_id, ROUND(SUM(payment_value), 2) AS
payment_value
FROM
    olist_payments
WHERE
    order_id IN (SELECT
        order_id
    FROM
        olist_orders
    WHERE
        order_status <> 'canceled'
        AND
        order_delivered_customer_date IS NOT NULL)
    GROUP BY order_id;
```

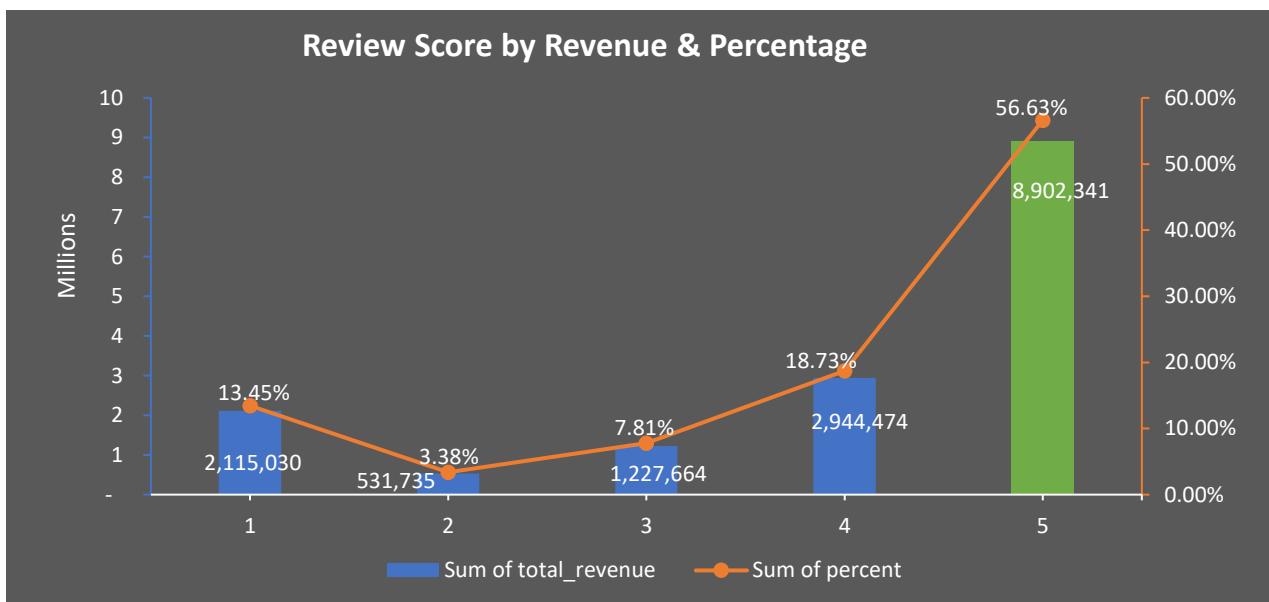
Let's join the two views to get the total revenue with respect to each review score

```
Total Revenue by Review Score

SELECT
    r.review_score,
    COALESCE(ROUND(SUM(p.payment_value), 0), 0) as
    total_payment_value,
    ROUND((COALESCE(ROUND(SUM(p.payment_value),
    0), 0) / SUM(SUM(p.payment_value)) OVER()) * 100,
    2) as percentage
FROM
    review as r
JOIN
    payment as p
    ON r.order_id = p.order_id
GROUP BY
    r.review_score
ORDER BY
    r.review_score DESC;
```

Output:

The review score of 5 recorded the highest revenue of R\$8,902,341 and the highest percentage proportion of 56.63%.



7. HOW MANY CUSTOMERS HAVE MADE REPEAT PURCHASES ON OLIST, AND WHAT PERCENTAGE OF TOTAL SALES DO THEY ACCOUNT FOR?

Solution:

```
Number of Return Customers

with return_customers as (
    select oc.customer_unique_id as rep_customer,
    count(distinct oo.order_id) as num_rep_customers
    from
        olist_orders as oo
    join
        olist_customers as oc on oo.customer_id =
    oc.customer_id
    where oo.order_status <> 'canceled'
    and oo.order_delivered_customer_date is not null
    group by rep_customer
    having count(oo.order_id) > 1
)
select count(*) as num_return_customers
from return_customers;
```

Output: We had a total of 2924 repeat customers on the Olist platform.

Total Number of Return Customers
2,924

LET'S FIND THE TOTAL REVENUE FROM THESE REPEAT CUSTOMERS ON THE OLIST PLATFORM

```
Revenue from Repeat Customers

WITH repeat_customers AS (
    SELECT
        oc.customer_unique_id AS rep_customers,
        SUM(opa.payment_value) AS total_revenue
    FROM
        olist_orders AS oo
    JOIN
        olist_customers AS oc ON oo.customer_id =
    oc.customer_id
    JOIN
        olist_payments AS opa ON oo.order_id =
    opa.order_id
    WHERE
        oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS
    NOT NULL
    GROUP BY rep_customers
    HAVING COUNT(DISTINCT oo.order_id) > 1
)
SELECT ROUND(SUM(total_revenue), 0) AS
revenue_rep_customers
FROM repeat_customers;
```

Output: The total revenue realized from repeat customers on the Olist platform is **R\$899,381**.

Total Revenue from Repeat Customers

R\$899,381

LET'S DETERMINE THE PERCENTAGE OF REPEAT CUSTOMERS' REVENUE IN COMPARISON TO TOTAL REVENUE ON THE OLIST PLATFORM.

Solution:

```
Percentage of Repeat Customers Revenue

SELECT
    ROUND((SUM(CASE WHEN
                    oc.customer_unique_id IN
                (SELECT
                    oc.customer_unique_id
                FROM
                    olist_orders AS oo
                JOIN olist_customers AS oc
                    ON oo.customer_id = oc.customer_id
                WHERE oo.order_status <> 'canceled' AND
                    oo.order_delivered_customer_date IS NOT NULL
                GROUP BY oc.customer_unique_id
                HAVING COUNT(oo.order_id) > 1)
            THEN
                opa.payment_value
            ELSE 0
        END) / SUM(opa.payment_value)) * 100,2)
    AS percent_rep_customer_rev
FROM
    olist_orders AS oo
JOIN
    olist_customers AS oc ON oo.customer_id =
        oc.customer_id
JOIN
    olist_payments AS opa ON oo.order_id =
        opa.order_id
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL;
```

Output:

The percentage of total revenue from repeat customers on the Olist platform was found to be just **5.67%**.

Percentage of Total Revenue from Repeat Customers

5.67%

8. WHAT IS THE AVERAGE CUSTOMER RATING FOR PRODUCTS SOLD ON OLIST, AND HOW DOES THIS IMPACT SALES PERFORMANCE?

LET'S FIND OUT THE AVERAGE REVIEW SCORE OR RATING FROM ORDERS MADE ON OLIST

Solution:

```
Average Review Score / Rating

SELECT
    ROUND(AVG(ore.review_score), 1) AS avg_review
FROM
    olist_orders AS oo
JOIN
    olist_reviews AS ore ON oo.order_id =
        ore.order_id
WHERE
    oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS
NOT NULL;
```

Output: The average review score is 4.1

Average Review Score / Rating
4.1

LET'S FIND OUT HOW THE AVERAGE REVIEW SCORES AFFECT SALES PERFORMANCE CONCERNING THE NUMBER OF PRODUCT ORDERS

Solution:

```
Avg. Review Score by No. of Orders(ProductOrders CTE)

WITH ProductOrders AS (
    SELECT
        oi.order_id,
        op.product_id,
        op.product_category_name_english AS product_name
    FROM
        olist_items AS oi
    JOIN
        olist_orders AS oo ON oi.order_id = oo.order_id
    JOIN
        olist_products AS op ON oi.product_id = op.product_id
    WHERE
        oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
),
```

```

Avg. Review Score by No. of Orders(ProductReviews CTE)

ProductReviews AS (
    SELECT
        oo.order_id,
        AVG(ore.review_score) AS avg_review_score
    FROM
        olist_reviews AS ore
    JOIN
        olist_orders AS oo ON ore.order_id = oo.order_id
    WHERE
        oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
    GROUP BY
        oo.order_id
)

```

```

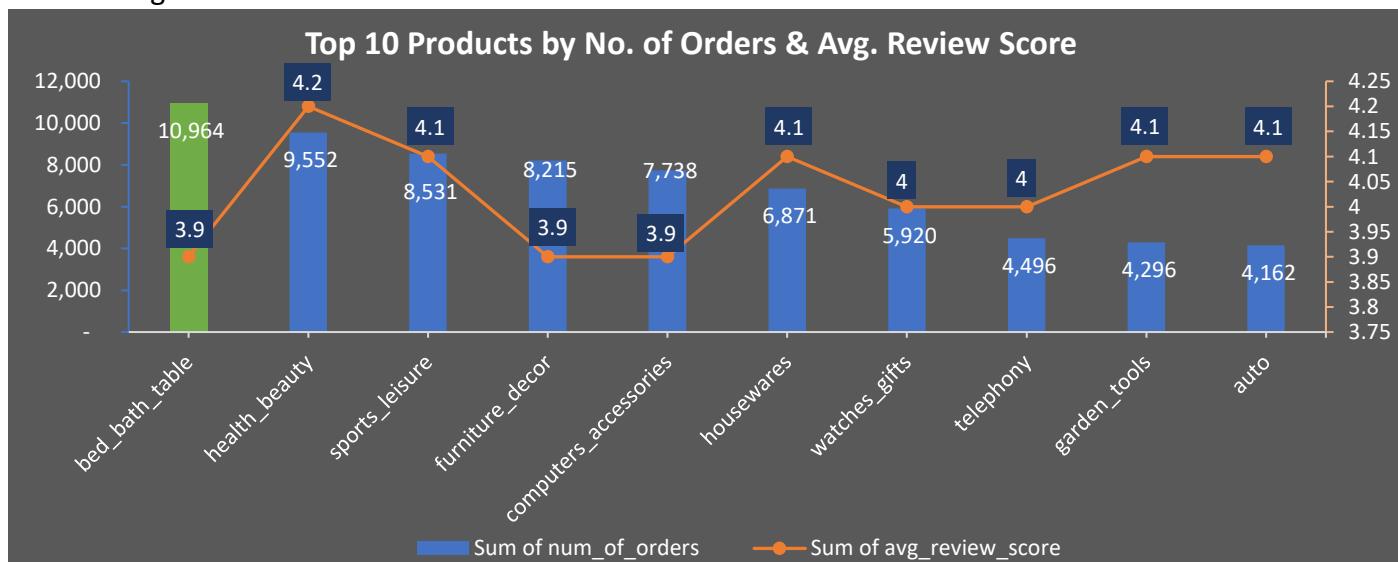
Avg. Review Score by Product name, No. of Orders & Rank

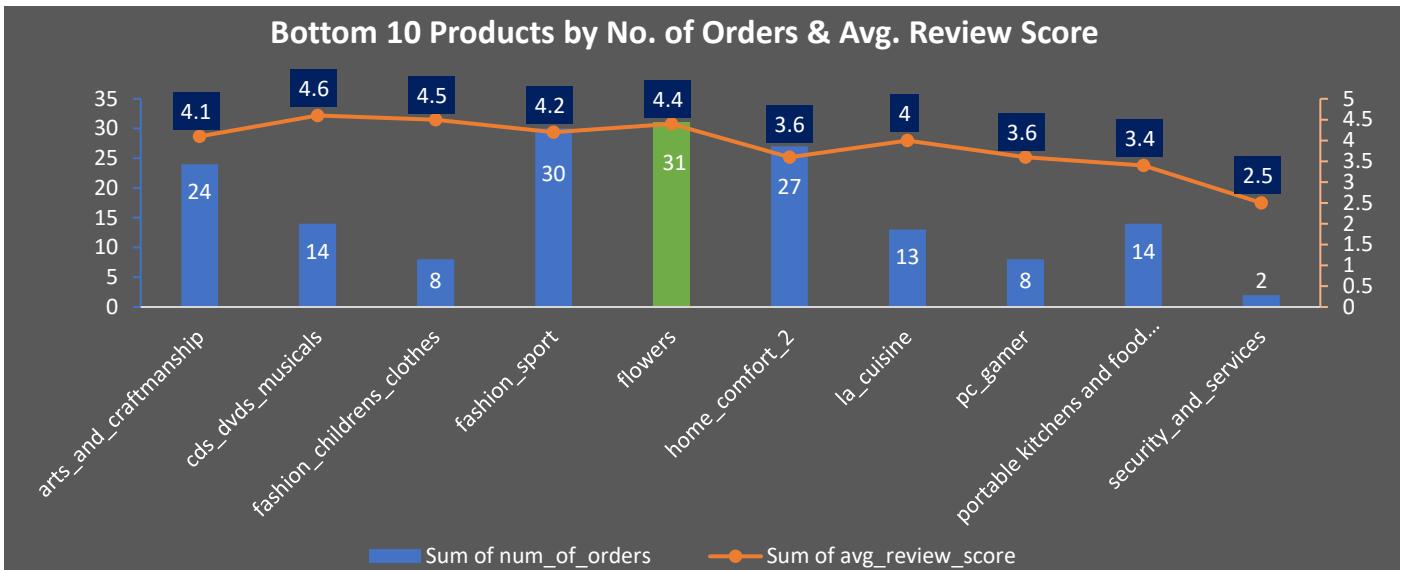
SELECT
    product_name,
    ROUND(AVG(pr.avg_review_score), 1) AS avg_review_score,
    COUNT(po.order_id) AS num_of_orders,
    RANK() OVER (ORDER BY COUNT(po.order_id) DESC) AS product_rank
FROM
    ProductOrders AS po
JOIN
    ProductReviews AS pr ON po.order_id = pr.order_id
GROUP BY
    product_name
ORDER BY
    product_rank;

```

Output:

Sales performance has been positive with the majority of product sales having an approximate average review score of 4.





9. WHAT IS THE ORDER CANCELLATION RATE PERCENTAGE?

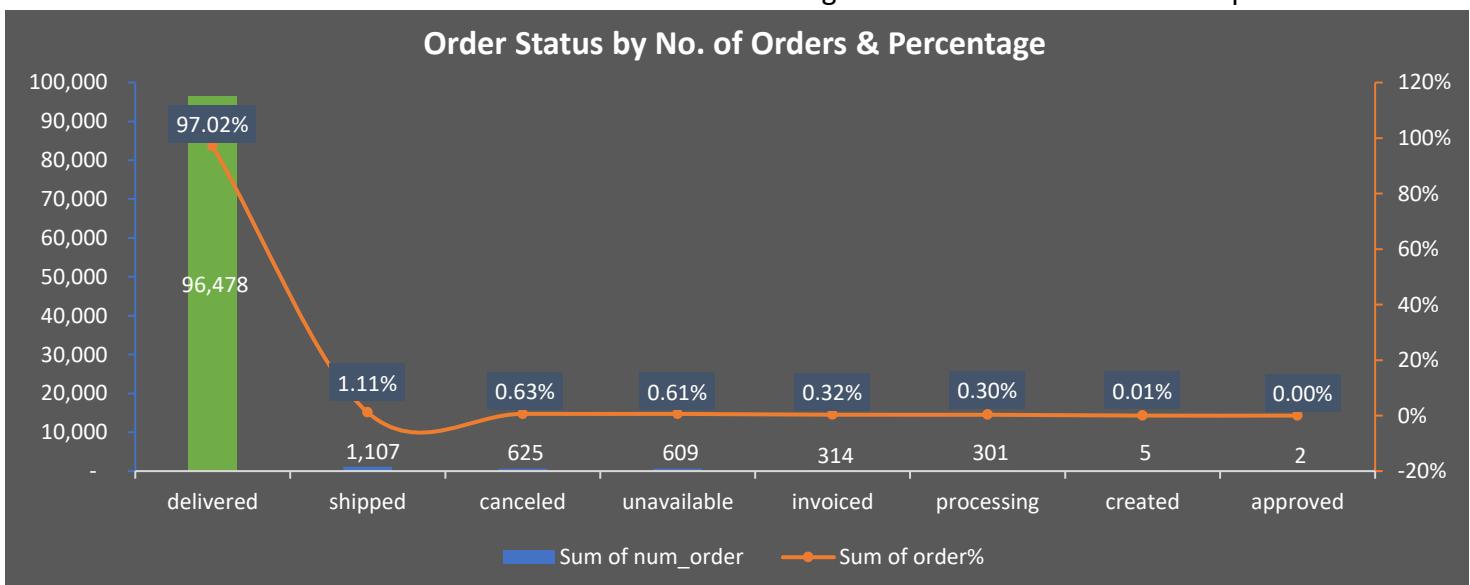
Solution:

```
Order Cancellation %

SELECT
    order_status,
    count(order_id) as num_order,
    round(100 * count(order_id) /
    sum(count(order_id)) over(), 2) as percentage
FROM
    olist_orders as oo
GROUP BY
    order_status;
```

Output:

The total number of **cancelled** orders is **625** accounting for **0.63%** of all orders on the platform.



11. WHICH PAYMENT METHODS ARE MOST COMMONLY USED BY CUSTOMERS, AND HOW DOES THIS VARY BY PRODUCT CATEGORY OR GEOLOCATION?

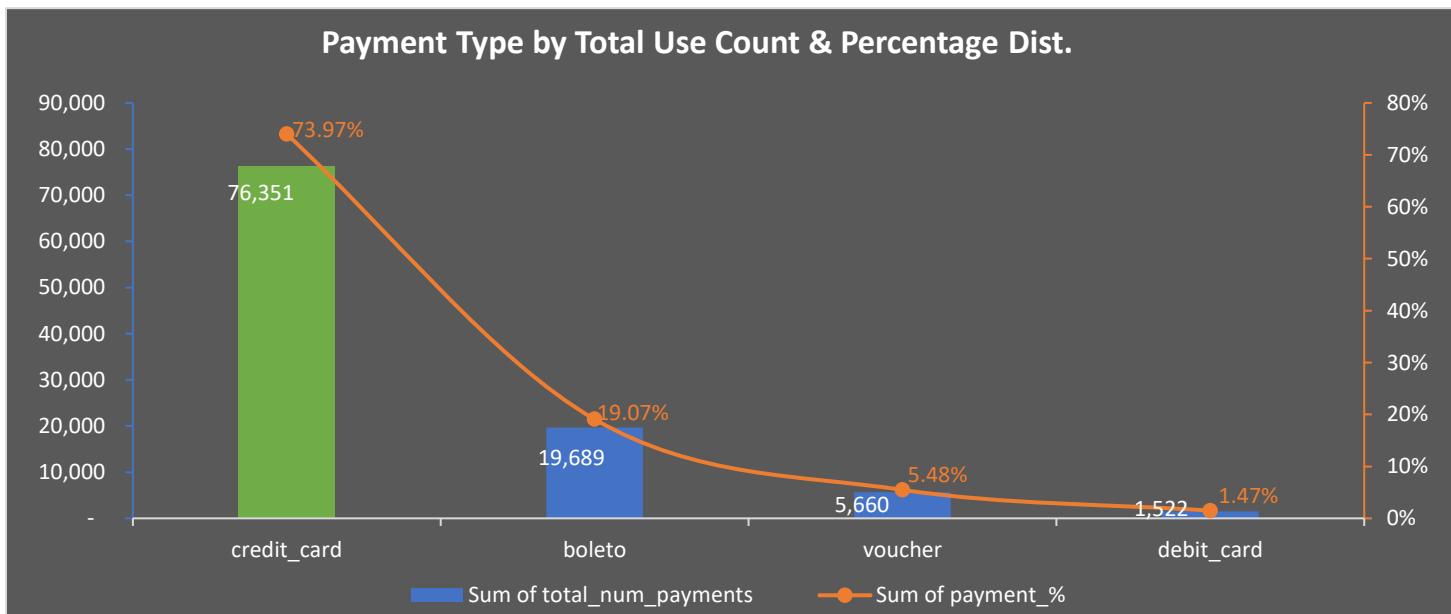
Solution:

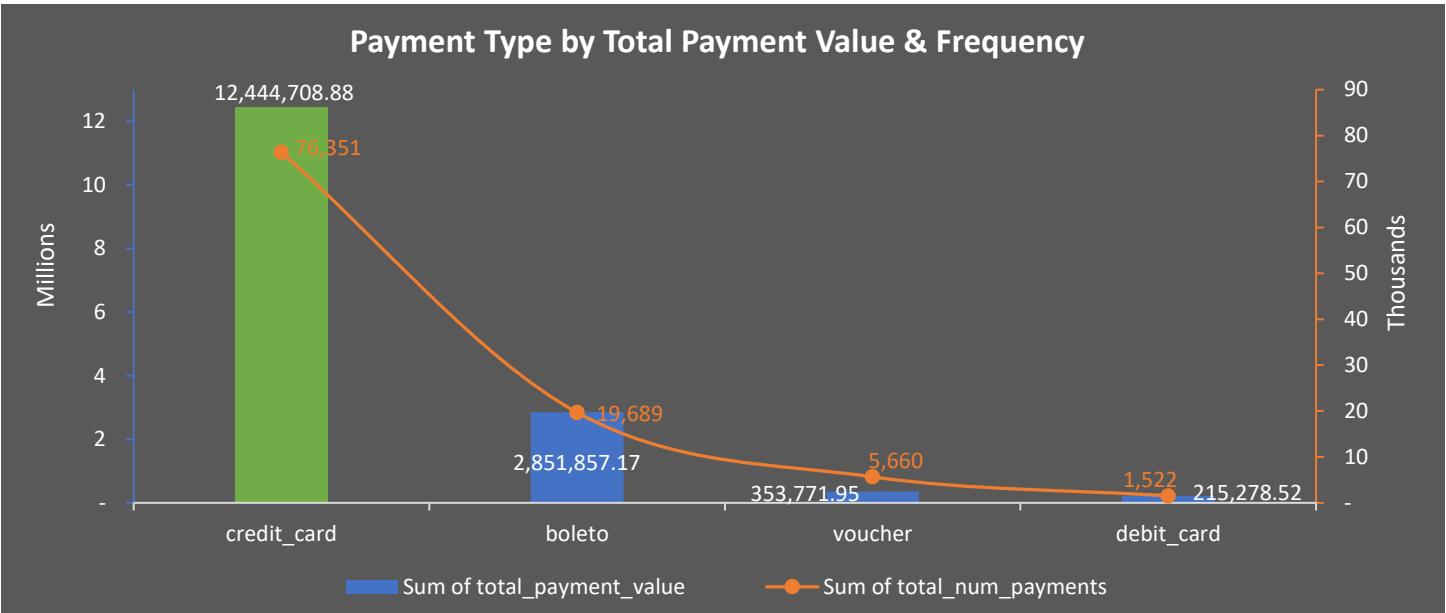
```
Most used payment type

SELECT
    payment_type,
    COUNT(*) AS total_num_payments,
    ROUND(SUM(payment_value), 2) AS total_payment_value,
    ROUND((COUNT(*) / total_payments.total_count) * 100, 2) AS percentage
FROM
    olist_payments AS opa
JOIN
    olist_orders AS oo ON oo.order_id = opa.order_id
JOIN
    (SELECT
        COUNT(*) AS total_count
    FROM
        olist_payments) AS total_payments
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY payment_type , total_payments.total_count;
```

Output:

Credit card payments were the highest with a total usage of **76,351** making up **73.49%** of all payment types on the platform and accounting for a total payment value of **R\$12,444,708.88**





LET'S LOOK AT HOW THESE PAYMENT METHODS VARIED ACROSS PRODUCT CATEGORIES.

Solution:

```

Most used payment type by Product

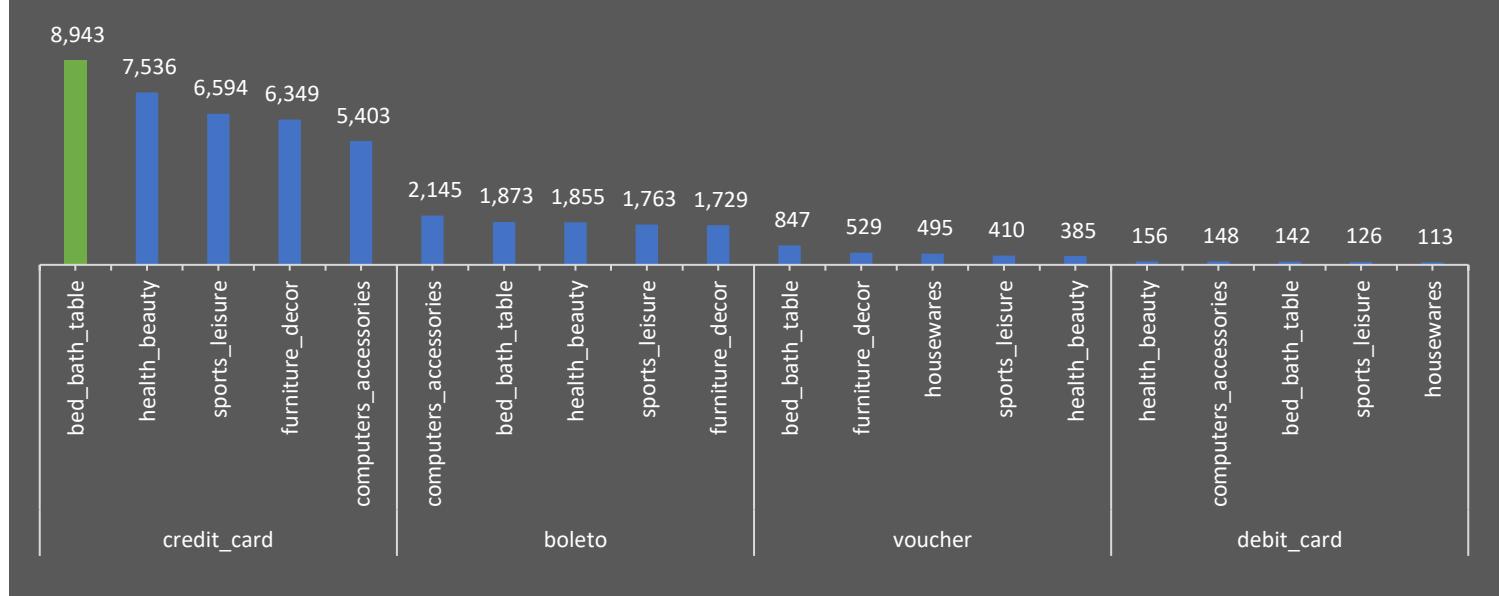
SELECT
    op.product_category_name_english AS product_name,
    opa.payment_type,
    COUNT(*) AS num_of_orders,
    RANK() OVER (ORDER BY COUNT(*) DESC) AS num_of_orders_rank
FROM
    olist_orders AS oo
JOIN
    olist_items AS oi ON oi.order_id = oo.order_id
JOIN
    olist_products AS op ON op.product_id = op.product_id
JOIN
    olist_payments AS opa ON opa.order_id = oo.order_id
WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
    product_name, payment_type
ORDER BY
    num_of_orders DESC;

```

Output:

For the top 5 product categories concerning the number of orders, the credit card payment method was the most used.

Top 5 Ordered Product Categories by Payment Type



LET'S LOOK AT THE NUMBER OF ORDERS BY GEOLOCATION AND THE PAYMENT METHODS MOST USED BY CUSTOMERS IN THESE LOCATIONS.

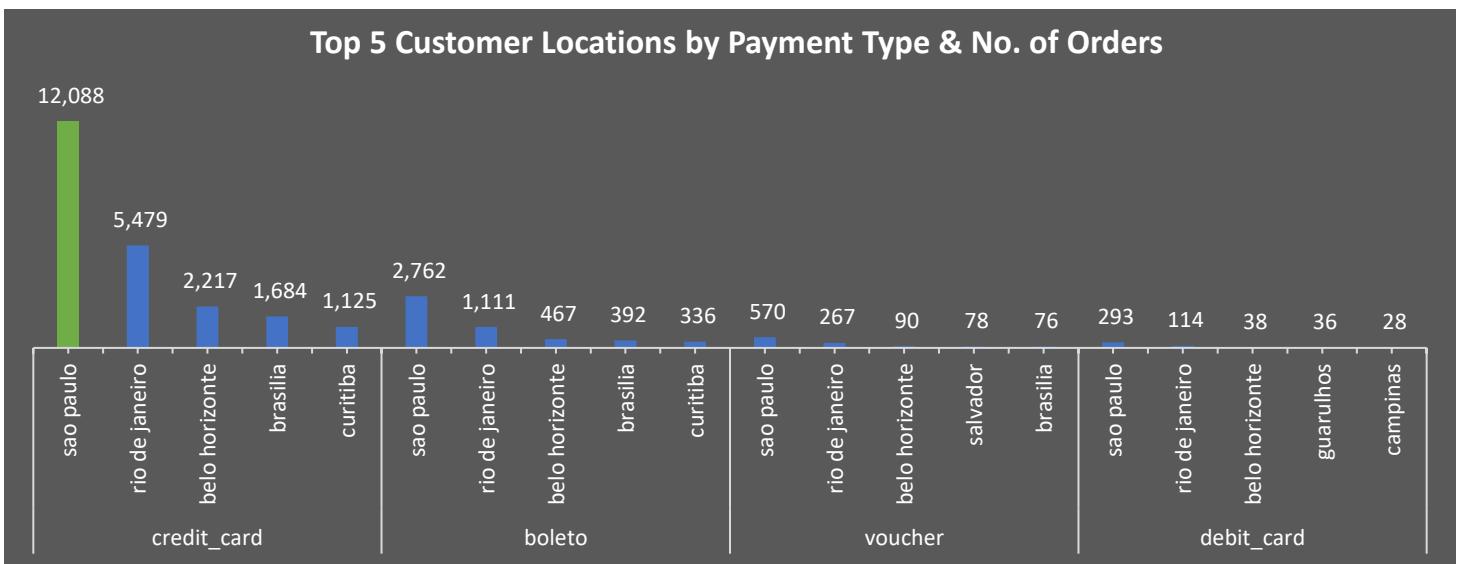
Solution:

```
Most used payment type by Customer Location & No. of Orders

SELECT oc.customer_city AS city,
       opa.payment_type AS payment_type,
       COUNT(DISTINCT opa.order_id) AS num_of_orders,
       RANK() OVER (ORDER BY COUNT(DISTINCT opa.order_id) DESC) AS
order_num_rank
FROM
       olist_payments AS opa
JOIN
       olist_orders AS oo ON oo.order_id = opa.order_id
JOIN
       olist_customers AS oc ON oc.customer_id = oo.customer_id
WHERE
       oo.order_status <> 'canceled'
       AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
       payment_type, city
ORDER BY
       num_of_orders DESC;
```

Output:

The highest number of orders came from **Sao Paulo** with a total order number of **12,088** and for these orders credit card payments were used.



12. WHICH PRODUCT CATEGORIES HAVE THE HIGHEST PROFIT MARGINS?

Solution:

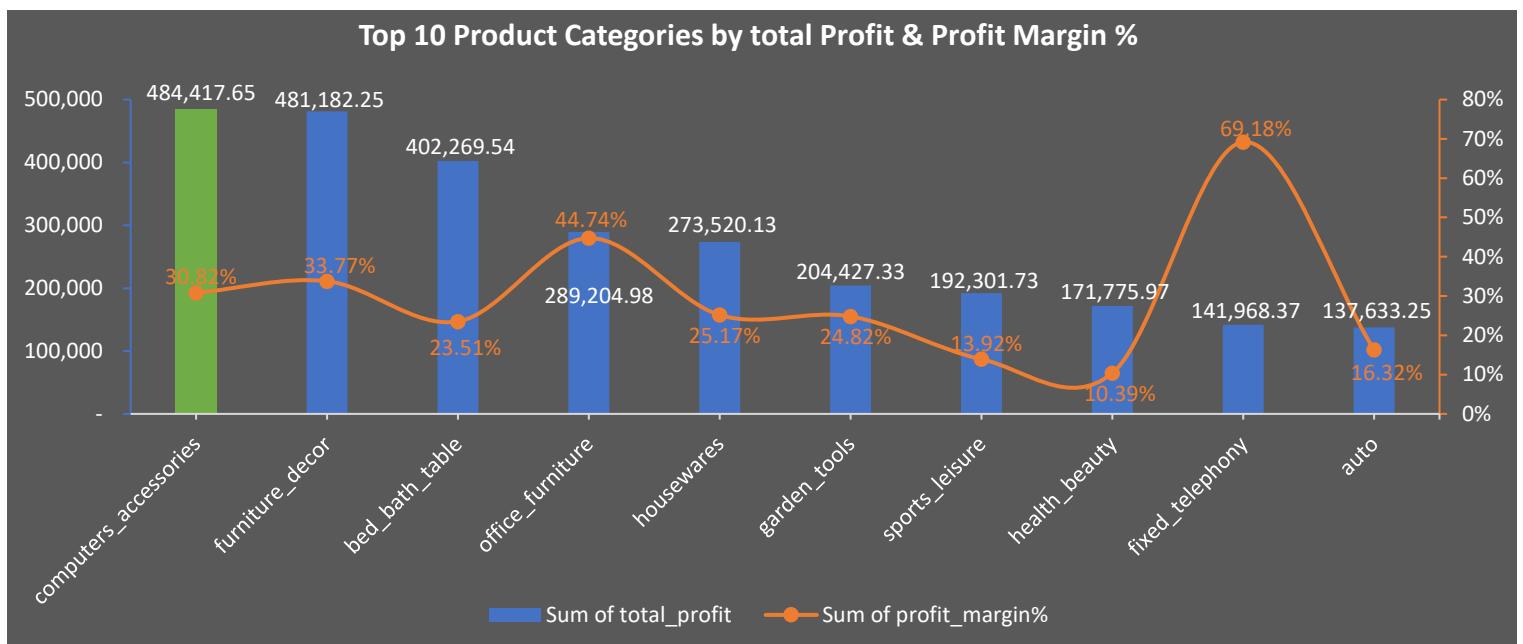
```
Highest Profit Margin by Product Category

SELECT
    op.product_category_name_english AS product_name,
    COUNT(DISTINCT oo.order_id) AS num_orders,
    ROUND(SUM(opa.payment_value), 2) AS total_payment,
    ROUND(SUM(oi.price + oi.freight_value), 2) AS total_cost,
    ROUND(SUM(opa.payment_value - (oi.price + oi.freight_value)), 2) AS
    total_profit,
    ROUND(((SUM(opa.payment_value) - SUM(oi.price + oi.freight_value)) /
    SUM(opa.payment_value)) * 100, 2) AS profit_margin_percentage
FROM
    olist_orders AS oo
    JOIN olist_items AS oi ON oo.order_id = oi.order_id
    JOIN olist_products AS op ON oi.product_id = op.product_id
    JOIN olist_payments AS opa ON oo.order_id = opa.order_id
WHERE
    oo.order_status <> 'canceled'
        AND oo.order_delivered_customer_date IS NOT NULL
GROUP BY
    op.product_category_name_english
ORDER BY
    total_profit DESC;
```

Output:

Computer_accessories recorded the highest profit of **R\$484,417.65** with a **30.82%** profit margin percentage, although this category did not have the highest number of orders.

Bed_bath_table recorded the highest number of orders **9,399** but recorded the third-highest profit of **R\$402,269.54** with a profit margin percentage of **23.51%**.

**13. WHICH GEOLOCATION (STATE) HAS THE HIGHEST CUSTOMER DENSITY? CALCULATE CUSTOMER RETENTION RATE (CRR) ACCORDING TO GEOLOCATIONS.****Solution:**

Customer Retention Rate (CRR) measures the percentage of existing customers who continue to use a product or service within a defined period. It provides insights into a company's effectiveness in building strong and lasting customer relationships.

```
● ● ● Location by Customer Density & CRR (return_customers CTE)

WITH return_customers AS (
  SELECT
    oc.customer_state AS state,
    COUNT(oc.customer_unique_id) AS ret_customers
  FROM
    olist_customers AS oc
  WHERE
    oc.customer_unique_id IN (
      SELECT
        unique_id
      FROM (
        SELECT
          oc.customer_unique_id AS unique_id,
          COUNT(DISTINCT oo.order_id)
        FROM
          olist_orders AS oo
        JOIN olist_customers AS oc ON oc.customer_id = oo.customer_id
        WHERE
          oo.order_status <> 'canceled'
          AND oo.order_delivered_customer_date IS NOT NULL
        GROUP BY
          oc.customer_unique_id
        HAVING
          COUNT(oo.order_id) > 1
        ORDER BY
          COUNT(DISTINCT oo.order_id) DESC
        ) AS rep_customers
    )
  GROUP BY
    oc.customer_state
  ),
  
```

```
● ● ● Location by Customer Density & CRR (total_customers CTE)

Total_customers AS (
  SELECT
    oc.customer_state AS state,
    COUNT(oc.customer_unique_id) AS tot_customer
  FROM
    olist_customers AS oc
    JOIN olist_orders AS oo ON
      oo.customer_id = oc.customer_id
  WHERE
    oo.order_status <> 'canceled'
    AND oo.order_delivered_customer_date IS NOT NULL
  GROUP BY
    oc.customer_state
  )
```



Location by Customer Density & CRR (final query)

```
SELECT
    rc.state AS state,
    tc.tot_customer AS number_of_customers,
    rc.ret_customers AS num_rep_customers,
    ROUND((ret_customers / tot_customer) * 100, 2) AS CRR
FROM
    return_customers AS rc
JOIN total_customers AS tc ON
    rc.state = tc.state
GROUP BY
    state
ORDER BY
    CRR DESC;
-- number_of_customers desc;
```

Output:

Sao Paulo (SP) had the highest customer density of **41,419**, of which **2,678** were repeat customers and recorded a Customer Retention Rate (CRR) of **6.47%**.

The state with the highest Customer Retention Rate (CRR) is **Acre (AC)** with a CRR of **9.88%**, with **81** customers and only **8** repeat customers.

