



# OLIST STORE ECOMMERCE ANALYSIS

## SQL PROJECT | MYSQL

# # OLIST STORE E COMMERCE ANALYSIS SQL PROJECT

- KPI 1: TOP CUSTOMERS BY TOTAL SPENT.
- KPI 2: TOP 10 BEST-SELLING PRODUCT CATEGORIES BY QUANTITY SOLD.
- KPI 3: ON-TIME DELIVERY RATE PERCENTAGE.
- KPI 4: AVERAGE REVIEW RATING BY PRODUCT CATEGORY.
- KPI 5: WEEKDAY VS WEEKEND (ORDER\_PURCHASE\_TIMESTAMP) PAYMENT STATISTICS.

# **# OLIST STORE E COMMERCE ANALYSIS SQL PROJECT**

- KPI 6: NUMBER OF ORDERS WITH REVIEW SCORE 5 AND PAYMENT TYPE AS CREDIT CARD.ALTER.**
- KPI 7: AVERAGE NUMBER OF DAYS TAKEN FOR ORDER\_DELIVERED\_CUSTOMER\_DATE FOR PET PET\_SHOP**
- KPI 8: AVERAGE PRICE AND PAYMENT VALUES FROM CUSTOMERS OF SAO PAULO CITY**
- KPI 9: RELATIONSHIP BETWEEN SHIPPING DAYS (ORDER\_DELIVERED\_CUSTOMER\_DATE - ORDER\_PURCHASE\_TIMESTAMP) VS REVIEW SCORES**

# -- KPI 1: TOP CUSTOMERS BY TOTAL SPENT.

```
• SELECT
    c.customer_id,
    COUNT(DISTINCT o.order_id) AS total_orders,
    SUM(p.payment_value) AS total_spent
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id
JOIN payments p ON o.order_id = p.order_id
GROUP BY c.customer_id
ORDER BY total_spent DESC
LIMIT 10;
```

customer_id	total_orders	total_spent
1617b1357756262bfa56ab541c47bc16	1	13664.08
ec5b2ba62e574342386871631fafd3fc	1	7274.88
c6e2731c5b391845f6800c97401a43a9	1	6929.31
f48d464a0baaea338cb25f816991ab1f	1	6922.21
3fd6777bbce08a352fddd04e4a7cc8f6	1	6726.66

## -- KPI 2: TOP 10 BEST-SELLING PRODUCT CATEGORIES BY QUANTITY SOLD.

```
SELECT
    pr.product_category_name,
    SUM(oi.order_item_id) AS total_quantity_sold
FROM order_items oi
JOIN products pr ON oi.product_id = pr.product_id
GROUP BY pr.product_category_name
ORDER BY total_quantity_sold DESC
LIMIT 10;
```

Result Grid			Filter Rows:
	product_category_name	total_quantity_sold	
▶	cama_mesa_banho	13655	
	moveis_decoracao	11540	
	beleza_saude	11081	
	esporte_lazer	9932	
	informatica_acessorios	9874	
	utilidades_domesticas	9051	
	relogios_presentes	6594	
	ferramentas_jardim	5874	
	telefonica	5161	
	automotivo	4881	

Result 46 x

## -- KPI 3: ON-TIME DELIVERY RATE PERCENTAGE.

SELECT

```
COUNT(*) AS total_orders,  
SUM(CASE WHEN order_delivered_customer_date <= order_estimated_delivery_date THEN 1 ELSE 0 END) AS on_time_deliveries,  
ROUND(  
    (SUM(CASE WHEN order_delivered_customer_date <= order_estimated_delivery_date THEN 1 ELSE 0 END) / COUNT(*)) * 100, 2  
) AS on_time_delivery_percentage  
FROM orders  
WHERE order_status = 'delivered';
```

	total_orders	on_time_deliveries	on_time_delivery_percentage
▶	96455	88630	91.89

## -- KPI 4: AVERAGE REVIEW RATING BY PRODUCT CATEGORY.

```
SELECT
    pr.product_category_name,
    AVG(rv.review_score) AS avg_review_rating,
    COUNT(rv.review_id) AS total_reviews
FROM reviews rv
JOIN order_items oi ON rv.order_id = oi.order_id
JOIN products pr ON oi.product_id = pr.product_id
GROUP BY pr.product_category_name
ORDER BY avg_review_rating DESC;
```

	product_category_name	avg_review_rating	total_reviews
	automotivo	4.065201815142107	4187
	utilidades_domesticas	4.056874095513749	6910
	construcao_ferramentas_construcao	4.055374592833876	921
	construcao_ferramentas_iluminacao	4.0508474576271185	295
	construcao_ferramentas_jardim	4.046218487394958	238
	ferramentas_jardim	4.042129629629629	4320
	eletronicos	4.03932993445011	2746
	bebidas	4.037533512064343	373

-- KPI 5: WEEKDAY Vs WEEKEND (ORDER\_PURCHASE\_TIMRSTAMP) PAYMENT STATISTICS.

```
SELECT
    kpi1.day_end,
    CONCAT(ROUND(kpi1.total_payments / (SELECT SUM(payment_value) FROM payments) * 100, 2), '%') AS percentage_values
FROM (
    SELECT
        ord.day_end,
        SUM(pmt.payment_value) AS total_payments
    FROM payments AS pmt
    JOIN (
        SELECT
            order_id,
            CASE
                WHEN WEEKDAY(order_purchase_timestamp) IN (5, 6) THEN 'Weekend'
                ELSE 'Weekday'
            END AS day_end
        FROM orders
    ) AS ord
    ON ord.order_id = pmt.order_id
    GROUP BY ord.day_end
) AS kpi1;
```

day_end	percentage_values
Weekday	74.38%
Weekend	21.94%



## -- KPI 6: NUMBER OF ORDERS WITH REVIEW SCORE 5 AND PAYMENT TYPE AS CREDIT CARD.ALTER.

```
SELECT
    COUNT(pmt.order_id) AS total_orders
FROM payments pmt
JOIN reviews rev
    ON pmt.order_id = rev.order_id
where
    rev.review_score = 5
    and pmt.payment_type = "credit_card";
```

total_orders
44011

## -- KPI 7: AVERAGE NUMBER OF DAYS TAKEN FOR ORDER\_DELIVERED\_CUSTOMER\_DATE FOR PET PET\_SHOP

```
SELECT
    p.product_category_name,
    ROUND(AVG(DATEDIFF(o.order_delivered_customer_date, o.order_purchase_timestamp)), 2) AS Avg_delivery_days
FROM orders AS o
JOIN order_items AS oi
    ON o.order_id = oi.order_id
JOIN products AS p
    ON oi.product_id = p.product_id
WHERE p.product_category_name = 'pet_shop'
    AND o.order_delivered_customer_date is not null
GROUP BY p.product_category_name;
```

product_category_name	Avg_delivery_days
pet_shop	11.17

## -- KPI 8: AVERAGE PRICE AND PAYMENT VALUES FROM CUSTOMERS OF SAO PAULO CITY

```
WITH orderItemsAvg AS (  
    SELECT  
        ROUND(AVG(oi.price)) AS avg_order_item_price  
    FROM order_items oi  
    JOIN orders ord ON oi.order_id = ord.order_id  
    JOIN customers cust ON ord.customer_id = cust.customer_id  
    WHERE cust.customer_city = 'colorado'  
)  
SELECT  
    (SELECT avg_order_item_price FROM orderItemsAvg) AS avg_order_item_price,  
    round(AVG(pmt.payment_value)) AS avg_payment_value  
FROM payments pmt  
JOIN orders ord ON pmt.order_id = ord.order_id  
JOIN customers cust ON ord.customer_id = cust.customer_id  
WHERE cust.customer_city = 'colorado';
```

avg_order_item_price	avg_payment_value
123	117

-- KPI 9:RELATIONSHIP BETWEEN SHIPPING DAYS (ORDER\_DELIVERED\_CUSTOMER\_DATE - ORDER\_PURCHASE\_TIMESTAMP) VS REVIEW SCORES

```
SELECT
    rew.review_score,
    ROUND(AVG(DATEDIFF(ord.order_delivered_customer_date, ord.order_purchase_timestamp)), 0) AS avg_shipping_days
FROM orders AS ord
JOIN reviews AS rew
    ON rew.order_id = ord.order_id
GROUP BY rew.review_score
ORDER BY rew.review_score;
```

review_score	avg_shipping_days
1	21
2	17
3	14
4	12
5	11

## CONCLUSION

**THROUGH THE USE OF SQL QUERIES, THE BRAZILIAN E-COMMERCE PUBLIC DATASET BY OLIST**

**WAS EXPLORED, CLEANED, AND ANALYZED TO EXTRACT MEANINGFUL PATTERNS. OPERATIONS LIKE FILTERING, SORTING, GROUPING, AND JOINING TABLES ENABLED DETAILED INSIGHTS INTO SALES TRENDS, CUSTOMER BEHAVIOR, AND PRODUCT PERFORMANCE. AGGREGATE FUNCTIONS AND CONDITIONS HELPED SUMMARIZE LARGE VOLUMES OF DATA EFFICIENTLY. THE ANALYSIS HIGHLIGHTS HOW SQL CAN TRANSFORM RAW DATA INTO CLEAR, ACTIONABLE INSIGHTS THAT SUPPORT BETTER DECISION-MAKING AND BUSINESS STRATEGY FORMULATION.**

The background is a dark navy blue. On the left side, there are several overlapping, semi-transparent blue geometric shapes, including rectangles and parallelograms, some of which are tilted. On the right side, there is a grid of small, light blue dots that fade out towards the center.

THANK YOU