OLIST BUSINESS ANALYSIS

SQL & MONGODB

SYEINRITA DEVI ANBEALAGAN

TABLE OF CONTENTS

INTRODUCTION

Olist is a significant Brazilian e-commerce network that connects vendors and customers across multiple product categories. It functions as a marketplace, allowing multiple merchants to post their products and facilitate transactions between them and clients. Olist provides a dependable and effective infrastructure for merchants to promote and sell their products, while also providing customers with a diverse selection and a smooth shopping experience. The objective of the company is to provide a convenient and dependable platform for customers to discover and purchase a wide range of products, while also allowing sellers to reach a bigger audience and build their companies. In the Brazilian market, Olist has established itself as a trusted brand, known for its rapid logistics, safe payment methods, and exceptional customer service.

The chosen dataset (link) , titled "Brazilian E-commerce Dataset by Olist," contains information on 100,000 orders placed between 2016 and 2018 on several Brazilian marketplaces. It offers detailed insights into several parts of the order process, such as order status, pricing, payment, freight performance, customer location, product attributes, and customer feedback. Olist hopes to get useful business insights, make better decisions, and improve overall performance by conducting data analysis on this dataset. The dataset is made up of nine.csv files, which are labelled "customer," "geolocation," "order items," "order payments," "order reviews," "orders," "products," "sellers," and "product category." These files contain a wealth of information that may be used to assess orders from multiple angles and unearth valuable insights for strategic decision-making.

ORGANISATIONAL STRUCTURE

To successfully manage its activities and maintain smooth coordination across different divisions, Olist adheres to a well-defined organisational structure. The corporation is divided into numerous important divisions, including:

1. Executive Leadership

   Top-level executives in charge of strategic decision-making, long-term planning, and overall corporate management.

2. Sales and marketing

   Responsible for introducing vendors to the platform, developing marketing tactics to improve customer interaction, and driving sales growth. This division also handles vendor partnerships and works with the product team to find trending and popular products.

3. Seller Management

   In charge of onboarding new sellers, managing seller relationships, and ensuring that Olist's policies and guidelines are followed. This division offers merchants assistance and training to help them optimise their product listings and increase their performance on the marketplace.

4. Operations and Logistics

   Manages the supply chain from beginning to end, including inventory management, order fulfilment, and delivery logistics. This division collaborates closely with sellers, suppliers, and shipping partners to ensure that orders are processed and delivered in a timely and effective manner.

5. Customer Support

   Provides great customer care and support to address inquiries, resolve issues, and ensure that customers have a positive buying experience. This branch handles consumer feedback, returns and refunds, and works constantly to improve customer happiness.

DATA USED IN THE BUSINESS

Olist captures and maintains client information such as names, addresses, phone numbers, and email addresses (olist_customers_dataset.csv). This information is used to personalise the shopping experience, track customer orders, and provide effective customer service.

- Olist uses geolocation data (olist_geolocation_dataset.csv) to understand the geographical distribution of customers and sellers. Zip codes, cities, states, and latitude-longitude coordinates are all part of this information. Geolocation data helps with logistics optimisation, targeting specific regions for marketing efforts, and analysing sales effectiveness across several sites.

- Order Items Data (olist_order_items_dataset.csv): This dataset offers information about the items that clients purchased. It contains information such as order IDs, product IDs, quantity, pricing, and seller IDs. Olist uses order item data to track sales, compute income, manage inventory, and analyse product popularity.

- Order Payments Data (olist_order_payments_dataset.csv): For each order, Olist stores payment information such as payment IDs, installments, payment methods, and transaction quantities. Olist can monitor financial transactions, track payment statuses, and generate sales reports using this data.

- Order Reviews Data (olist_order_reviews_dataset.csv): Customers can offer feedback and ratings for the things they buy. Review IDs, order IDs, ratings, review comments, and review timestamps are all included in the order reviews dataset. This information is used by Olist to track product satisfaction, gather feedback, and enhance seller and product quality.

- Orders Data (olist_orders_dataset.csv): This dataset contains detailed information about each client order, such as order IDs, customer IDs, order statuses, purchase timestamps, and delivery addresses. This information is used by Olist to track order processing, monitor delivery timings, and analyse consumer purchasing behaviour.

- Dataset of Products (olist_products_dataset.csv): This dataset offers data about the products that are offered on the Olist platform. Product IDs, names, descriptions, categories, and price range are all included. Olist uses product data to properly categorise and display products, provide product information to customers, and optimise product suggestions.

- Sellers Data (olist_sellers_dataset.csv): Olist stores information on sellers on the site, such as seller IDs, names, and locations. This information assists Olist in managing seller connections, tracking vendor performance, and facilitating seller-customer contact.

- Translations of Product Categories (product_category_name_translation.csv): This collection contains translations of the Olist platform's product categories. It contains the original product category names as well as their translated versions in English. This information enables Olist to cater to a wide consumer base and assure

SCENARIO

Olist has numerous years of industry experience and has seen constant growth in sales and an expanding customer base. However, as the e-commerce business evolves, the company recognises the need to closely review its processes and identify areas for improvement in order to keep its competitive edge.

According to DASH (2023), the global ecommerce market is expected to produce $6.3 trillion in 2023. To put it in context, if internet sales were a country, it would have the world's third-highest GDP, trailing only China and the United States. By the end of this year, the ecommerce market is expected to grow by 10.4%. With these figures in mind, Olist intends to capitalise on new possibilities and harness industry knowledge to achieve further growth. Olist can efficiently target its marketing efforts and encourage client loyalty by analysing customer behaviour and market trends, assuring a continuing gain in market share.

Besides, in today's cutthroat e-commerce environment, efficient logistics and delivery operations are critical. Olist understands the importance of simplifying its logistics procedures in order to improve delivery times and lower costs. Olist intends to obtain insights into order data, customer locations, and freight performance. List the products that are popular and profitable. With this information, the company may fine-tune its product management tactics, concentrate on high-demand items, and improve overall sales success. Furthermore, Olist recognises the importance of its sellers, who are critical to the company's success. Olist wishes to discover top-performing salespeople, recognise them appropriately, and improve its relationships with them in order to promote a collaborative and mutually advantageous environment.

Olist has decided to leverage the Brazilian Ecommerce Public Dataset that they have accumulated over the years to achieve these objectives. Olist intends to do sophisticated research, extract useful insights, and identify areas for improvement across many facets of company business by exploiting this vast dataset. Olist will be able to refine its strategy, optimise its processes, and increase its competitiveness in the dynamic Brazilian e-commerce sector thanks to this data-driven approach.

CORE BUSINESS INSIGHTS

1. Customer Segmentation: Analysing client data such as geography, order history, and preferences can aid in the identification of various customer segments. This segmentation can be used to efficiently target certain client groups, adjust marketing efforts, personalise the purchasing experience, and personalise the shopping experience.

2. Popular Product Categories: Olist can discover which product categories are most popular among customers by analysing the product dataset. This can be used to prioritise product procurement, optimise inventory management, and allocate marketing resources more effectively.

3. High-Demand Products: Analysing order item data will help you identify products with the highest demand and sales volume. Olist can concentrate on further advertising these products, assuring their availability, and investigating chances for cross-selling or upselling.

4. Preferred Payment Methods: Analysing order payment data can indicate the payment methods that customers most frequently utilise. This can assist Olist in optimising its payment processes, providing suitable payment choices, and improving the overall checkout experience.

5. Customer Satisfaction: Examining order reviews and ratings might provide useful information about customer satisfaction levels. To boost overall happiness and loyalty, Olist can detect recurring issues, resolve consumer concerns, and improve its customer care processes.

6. Seller Performance: Olist can identify top-performing sellers in terms of sales volume, customer satisfaction, and product quality by analysing seller data. This can assist Olist in recognising and rewarding high-performing sellers, improving connections with them, and encouraging collaboration for mutual success.

7. Regional Sales Patterns: Geolocation data can aid in the identification of sales patterns across various areas and cities. Based on regional sales trends, Olist may target marketing efforts, allocate inventory wisely, and optimise logistical operations.

8. Delivery Performance: Analysing order data and customer comments can provide insights into delivery performance, such as delivery times and customer happiness. Olist can discover areas for development, optimise logistics processes, and collaborate with dependable shipping suppliers to improve the delivery experience.

9. Order Status Analysis: Keeping track of order status data might assist in identifying bottlenecks or delays in the order fulfilment process. Olist can optimise order processing procedures, improve customer communication, and increase transparency throughout the order path.

10. Customer Lifetime Value (CLV) : Olist may compute its customers' lifetime value by analysing their order history. This indicator can aid in the prioritisation of customer retention efforts, the identification of high-value consumers, and the personalization of marketing campaigns to encourage client loyalty and repeat purchases.

11. Seasonal Trends: Analysing sales data over different time periods can reveal seasonal trends and patterns in customer behaviour. Olist can leverage this information to plan seasonal

promotions, adjust inventory levels, and optimise marketing campaigns to capitalise on peak seasons.

12. Price Sensitivity Analysis: By analysing customer purchase behaviour and price information from the products dataset, Olist can gain insights into price sensitivity across different customer segments and product categories. This knowledge can help in price optimization, promotional pricing strategies, and identifying opportunities for price adjustments.

ENTITY RELATIONSHIP DIAGRAM (ERD)

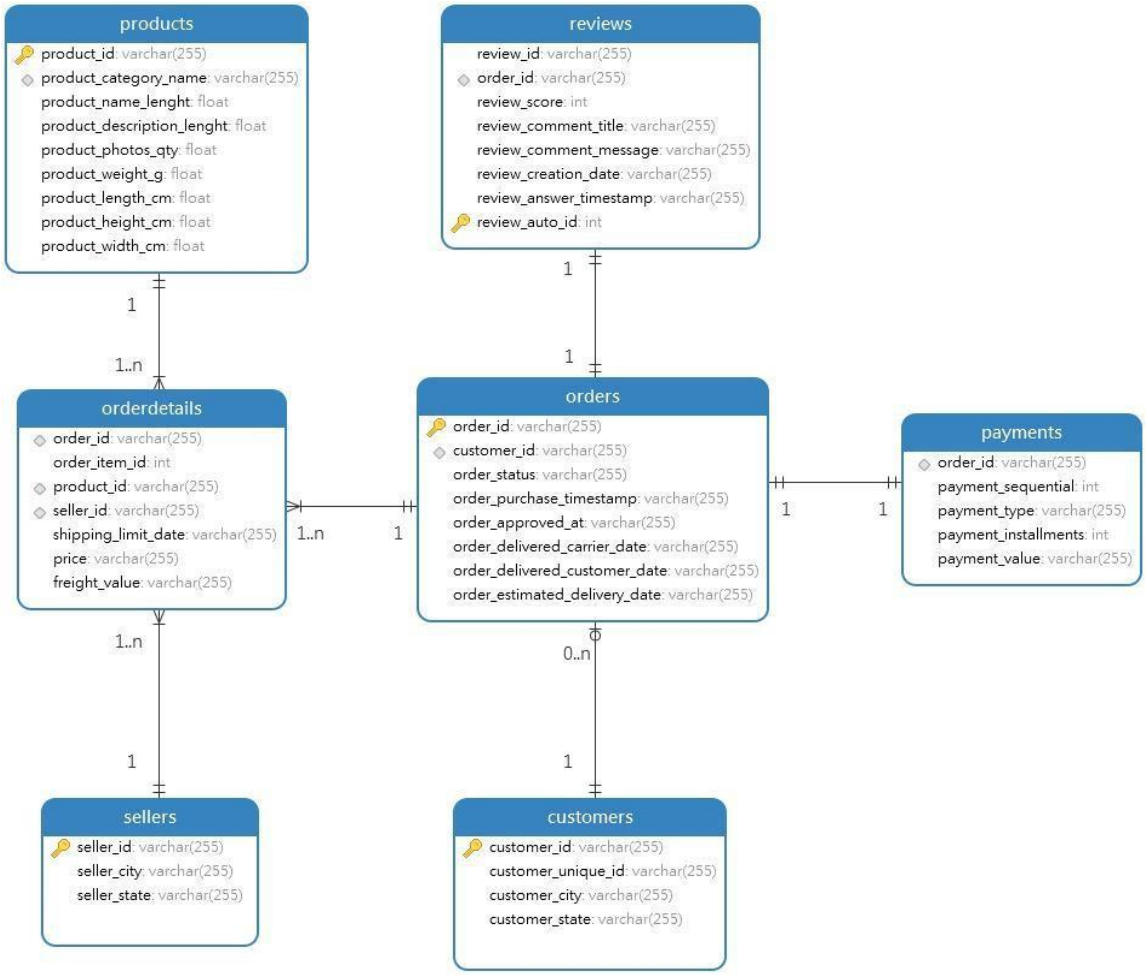


Figure 1 ERD

SQL QUERIES AND INDEX

INDEXES

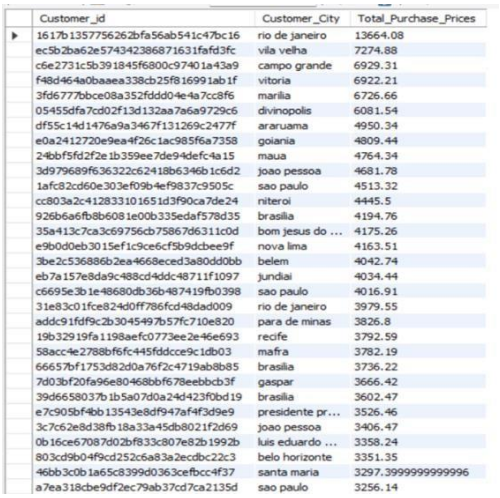
```
create index idx_customers_customer_id ON e_commerce.customers(customer_id);
create index idx_orders_customer_id ON e_commerce.orders(customer_id);
create index idx_orders_order_id ON e_commerce.orders(order_id);
create index idx_payments_order_id ON e_commerce.payments(order_id);
create index idx_products_product_id ON e_commerce.products(product_id);
create index idx_orderdetails_order_id ON e_commerce.orderdetails(order_id);
create index idx_orderdetails_product_id ON e_commerce.orderdetails(product_id);
```

BUSINESS INSIGHT 1

Which customer has the highest total price of purchases?

SQL QUERY 1

```
select c.customer_id as Customer_id, c.customer_city as Customer_City,
sum(pay.payment_value) as Total_Purchase_Prices
from customers c
join orders o on c.customer_id = o.customer_id
join payments pay on o.order_id = pay.order_id
group by c.customer_id, c.customer_city, o.order_id
order by Total_Purchase_Prices desc;
```



Customer_id	Customer_City	Total_Purchase_Prices
1617b1357756262bfa56ab541c47bc16	rio de janeiro	13664.08
ec5b2ba62e574342386871631fafd3fc	vila velha	7274.88
c6e2731c5b391845f6800c97401a43a9	campo grande	6929.31
f48d464a0baaea338cb25f816991ab1f	vitoria	6922.21
3fd6777bbce08a352fddd04e4a7cc8f6	marilia	6726.66
05455dfa7cd02f13d132aa7a6a9729c6	divinopolis	6081.54
df55c14d1476a9a3467f131269c2477f	araruama	4950.34
e0a2412720e9ea4f26c1ac985f6a7358	goiania	4809.44
24bbf5fd2f2e1b359ee7de94defc4a15	maua	4764.34
3d979689f636322c62418b6346b1c6d2	joao pessoa	4681.78
1afc82cd60e303ef09b4ef9837c9505c	sao paulo	4513.32
cc803a2c412833101651d3f90ca7de24	niteroi	4445.5
926b6a6fb8b6081e00b335edaf578d35	brasilia	4194.76
35a413c7ca3c69756cb75867d6311c0d	bom jesus do ...	4175.26
e9b0d0eb3015ef1c9ce6cf5b9dcbee9f	nova lima	4163.51
3be2c536886b2ea4668eced3a80dd0bb	belem	4042.74
eb7a157e8da9c488cd4ddc48711f1097	jundiai	4034.44
c6695e3b1e48680db36b487419fb0398	sao paulo	4016.91
31e83c01fce824d0ff786fcd48dad009	rio de janeiro	3979.55
addc91fdf9c2b3045497b57fc710e820	para de minas	3826.8
19b32919fa1198aefc0773ee2e46e693	recife	3792.59
58acc4e2788bf6fc445fddcce9c1db03	mafra	3782.19
66657bf1753d82d0a76f2c4719ab8b85	brasilia	3736.22
7d03bf20fa96e80468bbf678eebbcb3f	gaspar	3666.42
39d6658037b1b5a07d0a24d423f0bd19	brasilia	3602.47
e7c905bf4bb13543e8df947af4f3d9e9	presidente pr...	3526.46
3c7c62e8d38fb18a33a45db8021f2d69	joao pessoa	3406.47
0b16ce67087d02bf833c807e82b1992b	luis eduardo ...	3358.24
803cd9b04f9cd252c6a83a2ecdbc22c3	belo horizonte	3351.35
46bb3c0b1a65c8399d0363cefbcc4f37	santa maria	3297.3999999999996
a7ea318cbe9df2ec79ab37cd7ca2135d	sao paulo	3256.14

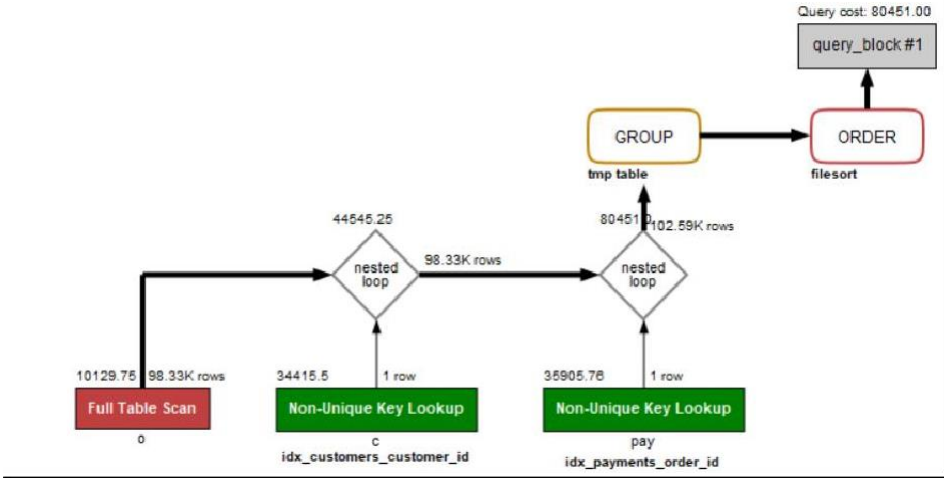
Query result Insight 1

INTERPRETATION OF INSIGHT 1 OUTPUT

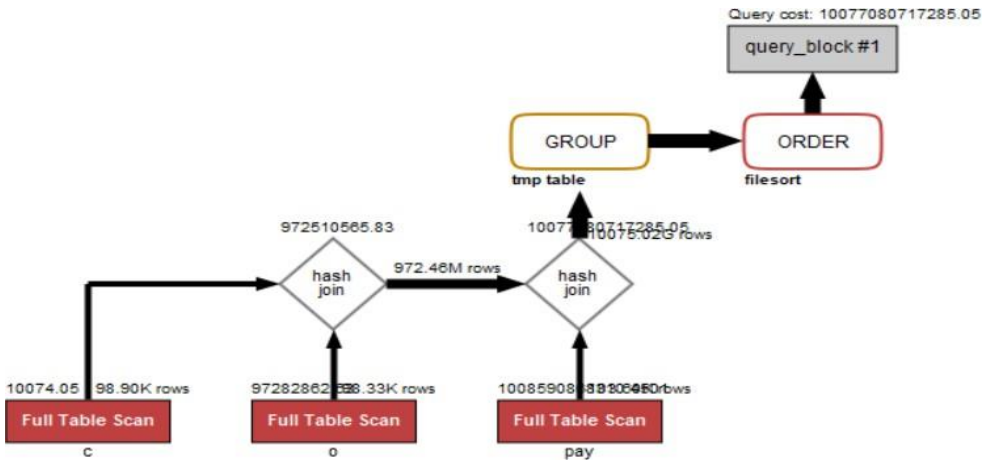
For Olist, the analysis of customer data yields useful insights. Customers' total purchase costs can range from 13664.08 to 3256.14 units. Significant purchase activity can be seen in important cities like Rio de Janeiro, Vila Velha, Campo Grande, and Vitoria. The spending habits of various customer segments, such as "f48d464a0baaea338cb25f81699ab1f1ab1f" and "3fd6777bbce08a352fdd04e4a7cc8f6," vary. To increase customer lifetime value (CLV), it is essential to identify high-value clients like "e0a2412720e9ea4f26c1ac985f6a7358." These insights guide Olist's marketing tactics, client retention initiatives, and overall company expansion.

COMPARISON OF WITH INDEXES AND WITHOUT INDEXES

Insight 1 with indexes:



Insight 1 without indexes:



BUSINESS INSIGHT 2

Which product categories have the highest sales?

SQL QUERY 2

```
select p.product_category_name as Product_Name, sum(od.price + od.freight_value) as Total_sales
from orderdetails od
join products p on od.product_id = p.product_id
join orders o on od.order_id = o.order_id
where o.order_status = "delivered"
group by p.product_category_name
order by Total_sales desc;
```

Product_Name	Total_sales
health_beauty	1412089.5299999977
watches_gifts	1264333.1199999969
bed_bath_table	1225209.259999999
sports_leisure	1118256.9100000008
computers_accessories	1032723.7699999965
furniture_decor	880329.919999993
housewares	758392.2500000012
cool_stuff	691680.8900000016
auto	669454.7499999993
garden_tools	567145.6799999983
toys	547061.059999999
baby	464766.6499999993
perfumery	443171.6300000048
telephony	379202.6200000058
office_furniture	335211.3600000011
stationery	269575.0500000005
pet_shop	250614.1999999996
computers	228349.75999999998
musical_instruments	202187.12000000008
electronics	200723.09000000084
small_appliances	198108.47000000038
fashion_bags_accesso...	179888.28999999948
luggage_accessories	168778.11999999994
consoles_games	167097.73000000085
construction_tools_co...	162382.01000000018
home_appliances_2	118208.52000000006
home_construction	95022.37000000008
home_appliances	93301.78
furniture_living_room	85045.07000000018
agro_industry_and_co...	76203.29999999997
home_confort	66438.21999999991

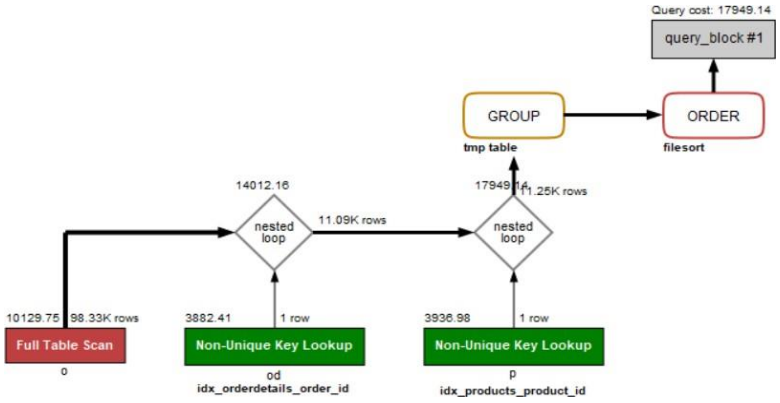
Query result Insight 2

INTERPRETATION OF INSIGHT 2 OUTPUT

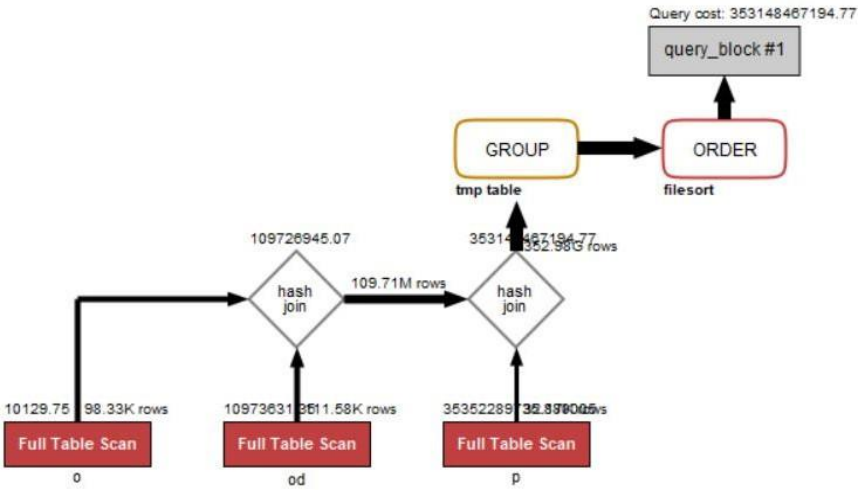
The top three product categories according to the analysis are "Health & Beauty," "Watches & Gifts," and "Bed & Bath Table." Olist can concentrate on extending its line of health and beauty products, collaborating with well-known brands, and launching targeted marketing campaigns with 1,412,089.53 units in total sales. Olist can explore partnerships with well-known watch brands, broaden its selection of gifts, and carry out promotional activities during busy gift-giving seasons thanks to its total sales of 1,264,333.12 units. Olist can broaden its selection of bed and bath products, improve the shopping experience in this market, and establish collaborations with home décor companies thanks to total sales of 1,225,209.26 units.

COMPARISON OF WITH INDEXES AND WITHOUT INDEXES

Insight 2 with indexes:



Insight 2 without indexes:



BUSINESS INSIGHT 3

Which products have the highest sale?

SQL QUERY 3

```
select p.product_id as Product_id, p.product_category_name as Product_Name, sum(od.price
+ od.freight_value) as Total_sales
from orderdetails od
join products p on od.product_id = p.product_id
group by p.product_id, Product_Name
order by Total_sales desc;
```

Product_id	Product_Name	Total_sales
bb50f2e236e5eea0100680137654686c	health_beauty	67606.10000000002
d1c427060a0f73f6b889a5c7c61f2ac4	computers_accessories	60976.03000000005
6cdd53843498f92890544667809f1595	health_beauty	59093.98999999994
99a4788cb24856965c36a24e339b6058	bed_bath_table	51071.59999999989
d6160fb7873f184099d9bc95e30376af	computers	50326.18
3dd2a17168ec895c781a9191c1e95ad7	computers_accessories	48212.22000000001
aca2eb7d00ea1a7b8ebd4e68314663af	furniture_decor	44820.76000000024
5f504b3a1c75b73d6151be81eb05bdc9	cool_stuff	41725.81000000001
25c38557cf793876c5abdd5931f922db	baby	40311.94999999999
53b36df67ebb7c41585e8d54d6772e08	watches_gifts	39957.93000000009
e0d64dcfaa3b6db5c54ca298ae101d05	watches_gifts	35344.09000000003
422879e10f46682990de24d770e7f83d	garden_tools	34201.26000000001
d285360f29ac7fd97640bf0baef03de0	watches_gifts	33607.57000000001
f1c7f353075ce59d8a6f3cf58f419c9c	bed_bath_table	33316.76000000002
7a10781637204d8d10485c71a6108a2e	watches_gifts	32790.73999999998
f819f0c84a64f02d3a5606ca95edd272	watches_gifts	30449.690000000006
588531f8ec37e7d5ff5b7b22ea0488f8	computers	29080.209999999985
389d119b48cf3043d311335e499d9c6b	garden_tools	28682.67999999996
a62e25e09e05e6faf31d90c6ec1aa3d1	watches_gifts	27998.30000000005
368c6c730842d78016ad823897a372db	garden_tools	27984.4
53759a2ecddad2bb87a079a1f1519f73	garden_tools	27268.22999999998
2b4609f8948be18874494203496bc318	health_beauty	26659.740000000005
16c4e87b98a9370a9cbc3a4658a3f45b	musical_instruments	26630.149999999994
52c80cedd4e90108bf4fa6a206ef6b03	garden_tools	25886.25999999999
19c91ef95d509ea33eda93495c4d3481	health_beauty	24326.250000000004
461f43be3bdf8844e65b62d9ac2c7a5a	watches_gifts	23215.830000000005
a5215a7a9f46c4185b12f38e9ddf2abc	computers	22411.020000000004
fd0065af7f09af4b82a0ca8f3eed1852	auto	22367.88
1dec4c88c685d5a07bf01dcb0f8bf9f8	auto	22228.510000000002
bc4cd4da98dd128c39bf0b8c2674032f	computers	22184.439999999995
c4baedd846ed09b85f78a781b522f126	garden_tools	20950.120000000006

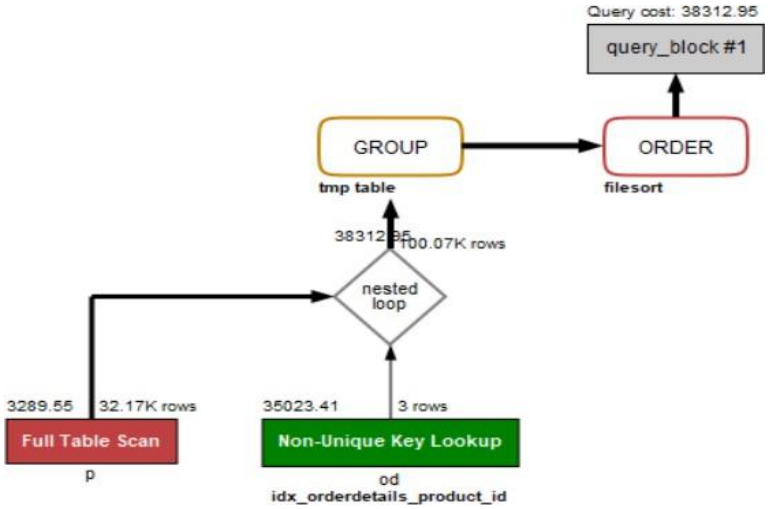
Query result Insight 3

INTERPRETATION OF INSIGHT 3 OUTPUT

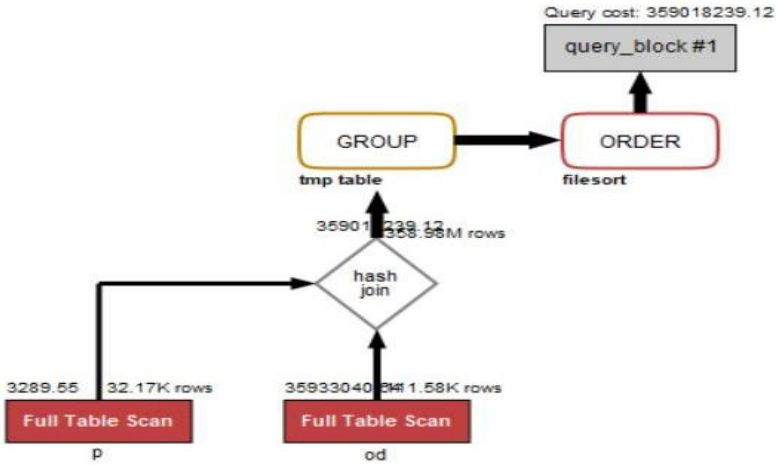
Olist can optimise its product assortment, raise customer satisfaction, and spur revenue growth by looking at the best-selling items and learning about customer preferences. Olist can also keep a close eye on these product categories, carry out market research to spot new trends, and maintain a competitive edge by consistently providing well-liked and in-demand goods. Products in the "Health & Beauty" category, for instance, have consistently had strong sales results. To increase sales in this market, Olist can concentrate on growing its selection of health and beauty products, collaborating with well-known brands, and launching targeted marketing campaigns.

COMPARISON OF USING INDEXES AND WITHOUT INDEXES

Insight 3 with indexes:



Insight 3 without indexes:



BUSINESS INSIGHT 4

Which payment methods are most commonly used?

SQL QUERY 4

```
select distinct(payment_type) as Payment_Types, count(payment_type) as Times_Of_Payments
from payments
group by payment_type;
```

	Payment_Types	Times_Of_Payments
▶	credit_card	76795
	boleto	19784
	voucher	5775
	debit_card	1529
	not_defined	3

Query result Insight 4

INTERPRETATION OF INSIGHT 4 OUTPUT

Olist customers' payment method usage analysis offers insightful data. With 76,795 transactions, credit card payments are the most popular and highlight the need for safe and practical credit card options. Boleto payments come in second with 19,784 transactions, demonstrating how crucial it is to streamline the payment process. With 5,775 transactions, voucher payments offer a chance for incentives to promote customer engagement. Although less common (1,529 transactions), debit card payments can be encouraged to increase the range of available payment methods. For accurate tracking, the small number of "Not Defined" payments must be addressed. With the aid of these insights, Olist is able to improve customer satisfaction, improve payment systems, and promote long-term growth.

BUSINESS INSIGHT 5

What is the number of individual ratings (review_score)?

SQL QUERY 5

```
SELECT
        review_score,
        COUNT(*) AS count
FROM
        reviews
GROUP BY
        review_score
ORDER BY
        count DESC;
```

	review_score	count
▶	5	57328
	4	19142
	1	11424
	3	8179
	2	3151

Query result Insight 5

INTERPRETATION OF INSIGHT 5 OUTPUT

The review ratings offered for Olist products provide insightful data on consumer satisfaction. With 57,328 reviews, the majority of customers have given it a rating of 5. This reveals a high degree of satisfaction and favourable comments about the products. With 19,142 reviews, a sizable portion of customers also gave a score of 4, supporting the generally positive sentiment. Olist can learn more about customer sentiment and perception of their products by understanding the distribution of review scores. Olist can continuously improve its products and services to meet and exceed customer expectations by actively listening to customer feedback and taking the necessary actions.

BUSINESS INSIGHT 5 (2)

What is the average user satisfaction in different states?

SQL QUERY 5 (2)

```
SELECT c.customer_state, AVG(r.review_score) AS average_score
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id
JOIN reviews r ON o.order_id = r.order_id
GROUP BY c.customer_state
ORDER BY average_score DESC;
```

customer_state	average_score
AP	4.1940
AM	4.1837
PR	4.1800
SP	4.1740
MG	4.1362
RS	4.1333
MS	4.1188
RN	4.1058
MT	4.1030
TO	4.0968
SC	4.0718
DF	4.0647
RO	4.0516
AC	4.0494
GO	4.0425
ES	4.0417
PB	4.0188
PE	4.0115
PI	3.9206
RJ	3.8750

Query result Insight 5(2)

INTERPRETATION OF INSIGHT 5(2) OUTPUT

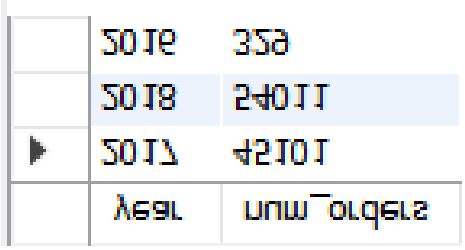
The number of 5 out of 5 ratings is 57,328, which is more than the sum of the 1-4 ratings, so more than 50% of users tend to give a full rating to shopping. AP states have the highest user satisfaction ratings with a mean of 4.19. RR states have the lowest user satisfaction ratings with a mean of 3.60. Olist can target marketing initiatives and respond to particular issues in various states by comprehending regional variations in customer satisfaction.

BUSINESS INSIGHT 5(3)

What is the total number of orders made by customers each year?

SQL QUERY 5(3)

```
WITH customer_orders AS (
    SELECT customer_id, COUNT(order_id) AS num_orders
    FROM Orders
    GROUP BY customer_id
)
SELECT YEAR(o.order_purchase_timestamp) AS year,
    SUM(co.num_orders) AS num_orders
FROM customer_orders co
JOIN Orders o ON co.customer_id = o.customer_id
GROUP BY YEAR(o.order_purchase_timestamp);
```



Query result Insight 5(3)

INTERPRETATION OF INSIGHT 5(3) OUTPUT

The information provided reveals how many orders Olist received annually. There were 329 orders in 2016, 45,101 in 2017, and 54,011 in 2018, which increased. Olist needs this data in order to monitor its development and assess its performance over time. Olist can evaluate the growth of its business and spot any patterns or trends in customer behaviour by examining the year-over-year increase in the number of orders. By using this information, Olist can more effectively manage its inventory, staff, and resources to meet the rising demand. It also sheds light on the efficiency of the marketing and promotion tactics Olist has employed.

BUSINESS INSIGHT 5(4)

What are the products with review score of 5 and total number of purchases ?

SQL QUERY 5(4)

```
SELECT product_category_name, p.product_id, COUNT(*)
FROM OrderDetails as i
INNER JOIN Products as p
ON p.product_id = i.product_id
INNER JOIN Reviews as r
ON r.order_id = i.order_id
WHERE r.review_score = 5
GROUP BY p.product_id;
```

product_category_name	product_id	COUNT(*)
computers_accessories	be0dbdc3d67d55727a65d4cd696ca73c	11
computers_accessories	d1c427060a0f73f6b889a5c7c61f2ac4	207
garden_tools	52c80cedd4e90108bf4fa6a206ef6b03	72
sports_leisure	3880d25d502b15b1de6fddc42ad1d67a	3
bed_bath_table	3152a0c0d93abeec99e4a6169aecc88b	1
toys	306e6b5dda3397e4a9312f94d82565b5	2
home_confort	35afc973633aaeb6b877ff57b2793310	82
toys	25e2023ed83352bde98dc1490d14c3d8	23
garden_tools	7c03e2562178adecc404571f7c52db4e	5
computers_accessories	6871a3c157d6f51697e887f3c3598479	2

Query result Insight 5(4)

INTERPRETATION OF INSIGHT 5(4) OUTPUT

The query offers useful details regarding the number of purchases made of highly rated products with a review score of 5. This aids Olist in locating top-rated goods so that it can concentrate on marketing and optimising them for higher sales and client satisfaction. It also emphasises how critical it is to address products with lower ratings in order to enhance customer satisfaction and address any problems. Olist is able to make data-driven decisions for product selection, marketing plans, and customer support thanks to the analysis of review scores and purchase information. This leads to increased customer loyalty and business growth.

BUSINESS INSIGHT 6

Who are the best performing sellers?

SQL QUERY 6

```
SELECT
    s.seller_id,
    COUNT(od.order_item_id) AS order_count,
    SUM(od.price) AS total_revenue
FROM OrderDetails od
JOIN Sellers s ON od.seller_id = s.seller_id
GROUP BY s.seller_id
ORDER BY total_revenue DESC;
```

seller_id	order_count	total_revenue
4869f7a5dfa277a7dca6462dcf3b52b2	1156	229472.6299999981
53243585a1d6dc2643021fd1853d8905	410	222776.04999999952
4a3ca9315b744ce9f8e9374361493884	1987	200472.9199999949
fa1c13f2614d7b5c4749cbc52fecda94	586	194042.02999999846
7c67e1448b00f6e969d365cea6b010ab	1364	187923.8899999995
7e93a43ef30c4f03f38b393420bc753a	340	176431.86999999982
da8622b14eb17ae2831f4ac5b9dab84a	1551	160236.56999999538
7a67c85e85bb2ce8582c35f2203ad736	1171	141745.53000000177
1025f0e2d44d7041d6cf58b6550e0bfa	1428	138968.5499999995
955fee9216a65b617aa5c0531780ce60	1499	135171.69999999914

Query result Insight 6

INTERPRETATION OF INSIGHT 6 OUTPUT

According to their respective seller IDs (4869f7a5dfa277a7dca64, 53243585a1d6dc264302, and 4a3ca9315b744ce9f8e9), the top three sellers on Olist have impressive total sales. Their success offers Olist insightful information and advantages. In accordance with their strategies, Olist can optimise the platform, set performance benchmarks, provide targeted support and resources, and share their expertise with other sellers. Olist can build a thriving ecosystem that promotes business growth, guarantees customer satisfaction, and propels platform success overall by capitalising on the success of these top sellers.

BUSINESS INSIGHT 7

What are the most profitable regions?

SQL QUERY 7

```
SELECT
    customer_state,
    COUNT(DISTINCT Customers.customer_id) AS customer_count,
    SUM(OrderDetails.price) AS total_sales
FROM Orders
JOIN Customers ON Orders.customer_id = Customers.customer_id
JOIN OrderDetails ON Orders.order_id = OrderDetails.order_id
GROUP BY customer_state
ORDER BY total_sales DESC;
```

customer_state	customer_count	total_sales
SP	41375	5202955.0500015225
RJ	12762	1824092.6699998123
MG	11544	1585308.0299998762
RS	5432	750304.0200000224
PR	4998	683083.7600000183
SC	3612	520553.34000000806
BA	3358	511349.99000000797
DF	2125	302603.9399999968
GO	2007	294591.9499999975
ES	2025	275037.30999999586

Query result Insight 7

INTERPRETATION OF INSIGHT 7(2) OUTPUT

Key insights for Olist are revealed by the sales volume data for various customer cities and states. The largest city in Brazil, Sao Paulo, is in first place with a sales volume of 15,540. Rio de Janeiro is second with 6,882 and Belo Horizonte is third with 2,773. These cities serve as sizeable markets for Olist and show the potential for future expansion and market encroachment. Olist can better serve its customers, improve customer experiences, and increase its presence in important areas by concentrating on these cities. Additionally, Olist is able to allocate resources wisely, spot potential growth areas, and focus its efforts to maximise sales and customer satisfaction by knowing the sales volume in various cities and states.

BUSINESS INSIGHT 8

What is the number of delivery days for each order?

SQL QUERY 8

```
SELECT
        orders.order_id,
        customers.customer_city,
        customers.customer_state,
        TIMESTAMPDIFF(DAY,orders.order_purchase_timestamp,
orders.order_delivered_customer_date ) AS delivery_days,
        reviews.review_score
FROM
        orders
        JOIN reviews ON orders.order_id = reviews.order_id
        JOIN customers ON orders.customer_id = customers.customer_id
WHERE
        orders.order_status = 'delivered'
        AND orders.order_delivered_customer_date IS NOT NULL
ORDER BY
        delivery_days DESC;
```

order_id	customer_city	customer_state	delivery_days	review_score
1b3190b2dfa9d789e1f14c05b647a14a	rio de janeiro	RJ	208	2
440d0d17af552815d15a9e41abe49359	belem	PA	195	1
2fb597c2f772eca01b1f5c561bf6cc7b	teresina	PI	194	4
0f4519c5f1c541ddec9f21b3bddd533a	teresina	PI	194	4
285ab9426d6982034523a855f55a885e	lagarto	SE	194	1
47b40429ed8cce3aee9199792275433f	salto	SP	191	1
2fe324febf907e3ea3f2aa9650869fa5	paulinia	SP	189	1
2d7561026d542c8dbd8f0daeadf67a43	aracaju	SE	188	3
437222e3fd1b07396f1d9ba8c15fba59	macapa	AP	187	5
c27815f7e3dd0b926b58552628481575	perdizes	MG	187	3
dfe5f68118c2576143240b8d78e5940a	teutonia	RS	186	4
6e82dcfb5eada6283dba34f164e636f5	santa maria	RS	182	5
2ba1366baecad3c3536f27546d129017	formosa	GO	181	1
d24e8541128cea179a11a65176e0a96f	sao bernardo...	SP	175	4
3566eabb132f8d64741ae7b921bbd10e	currais novos	RN	174	3

Query result Insight 8

INTERPRETATION OF INSIGHT 8 OUTPUT

The result indicates that the order with the longest delivery time was 1b3190b2dfa9d789e1f14c05b647a14a from rio de janeiro, RJ, which took a total of 208 days to deliver and was rated 2 out of 5. This outcome suggests that the devilry duration does have an impact on the customers' reviews. Olist can thus spot any patterns or trends by comparing the review scores with the corresponding delivery days. Olist can learn from this analysis whether quicker or slower delivery times have an effect on review and customer satisfaction ratings. Olist can learn more about what influences customers' positive or negative experiences by examining customer feedback and review scores for orders with different delivery times. This data can help Olist improve its delivery methods, streamline order fulfilment, and help customers set reasonable delivery expectations.

BUSINESS INSIGHT 9

What is the number of different order status categories and their percentages?

SQL QUERY 9

```
SELECT order_status, COUNT(order_id) AS order_count,
ROUND(COUNT(order_id)/(SELECT COUNT(order_id) from orders)*100,2) AS percentage
FROM Orders
GROUP BY order_status;
```

order_status	order_count	percentage
delivered	96478	97.02
unavailable	609	0.61
shipped	1107	1.11
canceled	625	0.63
invoiced	314	0.32
processing	301	0.30
approved	2	0.00
created	5	0.01

Query result Insight 9

INTERPRETATION OF INSIGHT 9 OUTPUT

The provided output provides a summary of the percentage distribution of order statuses and their corresponding order counts. Orders that have been "delivered" fall under this category make up 97.02% of the total. This shows that a sizable percentage of orders have been successfully filled and delivered to clients. A small portion of orders, though, are marked as "unavailable," "shipped," "cancelled," "invoiced," "processing," "approved," and "created." With "unavailable" and "cancelled" having the lowest percentages at 0.61% and 0.63%, respectively, these statuses represent different stages in the order process. Olist can track the progress of orders and spot potential problems in the fulfilment process by keeping an eye on the distribution of order statuses. Olist can streamline its operations, increase the effectiveness of order processing, address customer concerns, and ultimately improve the overall order fulfilment experience by analysing this data.

BUSINESS INSIGHT 9 (2)

Is there are any relationships between the order status and the rating for the orders?

SQL QUERY 9 (2)

```
SELECT order_status,
SUM (IF (review_score = 1,1,0)) AS score_1,
SUM(IF (review_score = 2,1,0)) AS score_2,
SUM (IF (review_score = 3,1,0)) AS score_3,
SUM(IF (review_score = 4,1,0)) AS score_4,
SUM(IF (review_score = 5,1,0)) AS score_5
FROM (SELECT order_status, review_score
FROM Orders o JOIN Reviews re
ON o.order_id = re.order_id) a
GROUP BY order_status
ORDER BY order_status;
```

	order_status	score_1	score_2	score_3	score_4	score_5
▶	approved	1	0	0	1	0
	canceled	422	44	48	26	69
	created	2	0	0	0	1
	delivered	9406	2941	7961	18987	57066
	invoiced	230	26	16	15	26
	processing	256	18	9	6	7
	shipped	644	79	110	87	123
	unavailable	463	43	35	20	36

Query result Insight 9(2)

INTERPRETATION OF INSIGHT 9(2) OUTPUT

Olist can learn a lot from the output on order status and associated scores (score_1 to score_5). Olist can evaluate performance and customer satisfaction at various points in the order process by analysing this data. High "delivered" status scores reflect effective delivery and fulfilment procedures and reflect positive customer experiences. Lower ratings for the "cancelled" or "unavailable" statuses, on the other hand, point out areas that could use improvement, such as lowering cancellation rates and raising product availability. This enables Olist to pinpoint specific areas that need improvement and attention, resulting in higher order conversion rates, higher customer satisfaction, and perhaps even fewer cancellations.

BUSINESS INSIGHT 10

What is overall revenue generated by individual customers?

SQL QUERY 10

```
SELECT customer_id, SUM(payment_value) AS total_payment
FROM Payments
GROUP BY customer_id;
```

customer_id	total_payment
00012a2ce6f8dcda20d059ce98491703	114.74
000161a058600d5901f007fab4c27140	67.41
0001fd6190edaaf884bcaf3d49edf079	195.42
0002414f95344307404f0ace7a26f1d5	179.35
000379cdec625522490c315e70c7a9fb	107.01
0004164d20a9e969af783496f3408652	71.8
000419c5494106c306a97b5635748086	49.4
00046a560d407e99b969756e0b10f282	166.59
00050bf6e01e69d5c0fd612f1bcfb69c	85.23

Query result Insight 10

INTERPRETATION OF INSIGHT 10 OUTPUT

These results show how much money each customer has spent overall. Olist can identify high-value customers and learn more about consumer spending habits by analysing this data. It enables Olist to divide customers into groups based on the amounts of their payments and adjust marketing plans accordingly. For customer retention and loyalty, it is essential to comprehend customer payment behaviour. Olist can use this information to provide tailored promotions, discounts, or rewards to customers who make larger payments, improving the customer experience and enticing them to make additional purchases.

BUSINESS INSIGHT 11

What is the customer behaviour over time and the pattern of it?

SQL QUERY 11

```
SELECT MONTH (order_purchase_timestamp) as month, YEAR (order_purchase_timestamp)
as year, SUM(price order_item_id) as total_sales
FROM OrderDetails
JOIN Orders ON OrderDetails.order_id = Orders.order_id
GROUP BY MONTH(order_purchase_timestamp), YEAR (order purchase_timestamp)
ORDER BY year, nonth;
```

	month	year	total_sales
▶	9	2016	435.23
	10	2016	56103.790000000154
	12	2016	10.9
	1	2017	142077.29999999932
	2	2017	269786.65999999637
	3	2017	412016.4300000014
	4	2017	399336.7900000022
	5	2017	562388.0900000099
	6	2017	471648.7200000052
	7	2017	558035.6000000099

Query result Insight 11

INTERPRETATION OF INSIGHT 11 OUTPUT

The provided sales data is organised by month and year, enabling a thorough examination of previous sales trends. Olist can learn a lot about the dynamics of their business by tracking the monthly sales figures over time. This analysis aids in the discovery of recurring patterns like seasonal peaks or regular monthly trends. Olist can identify distinctive patterns and determine whether particular months have consistently higher or lower sales by comparing sales for the same month across various years. Olist is now able to make data-driven decisions in areas like resource allocation, marketing initiatives, and inventory management. Olist can effectively maximise revenue and more precisely meet customer demands by optimising their operations and strategies based on a thorough understanding of sales patterns.

BUSINESS INSIGHT 11 (2)

What is popular time of the day where the shopping is done?

SQL QUERY 11 (2)

```
SELECT
CASE
WHEN HOUR(o.order_purchase_timestamp) BETWEEN 0 AND 5 THEN 'Dawn'
WHEN HOUR(o.order_purchase_timestamp) BETWEEN 6 AND 11 THEN 'Morning'
WHEN HOUR(o.order_purchase_timestamp) BETWEEN 12 AND 17 THEN 'Afternoon'
ELSE 'Night'
END AS time_of_day,
COUNT(*) as num_orders
FROM Orders o
GROUP BY time_of_day
ORDER BY num_orders DESC;
```

time_of_day	num_orders
Afternoon	38361
Night	34100
Morning	22240
Dawn	4740

Query result Insight 11(2)

INTERPRETATION OF INSIGHT 11(2) OUTPUT

The ability to gain insights into customer behaviour and preferences in relation to when they place orders makes this information valuable to Olist. Olist can efficiently meet customer demands during peak times by optimising their operations, such as staffing and order fulfilment, by analysing the order distribution across these time periods. enables Olist to adjust their marketing plans and promotional efforts to particular times when it is likely that customer engagement will be higher. Olist is able to improve the overall customer experience and promote business growth by better understanding customer patterns and tendencies regarding the time of day they make purchases. For instance, Olist can take advantage of the fact that

customers typically make more purchases in the afternoon based on the data by launching new products and running targeted promotions during that time to further engage and attract customers.

BUSINESS INSIGHT 12

How sensitive are the customers to price?

SQL QUERY 12

```
SELECT product_category_name, AVG(price) AS average_price, COUNT(*) AS order_count
FROM Products
JOIN OrderDetails ON Products.product_id = OrderDetails.product_id
GROUP BY product_category_name;
```

product_category_name	average_price	order_count
perfumery	116.73731207955527	3419
auto	139.95752302243392	4235
bed_bath_table	93.296327485387	11115
housewares	90.78814761631563	6964
watches_gifts	201.1359839759635	5991
cool_stuff	167.35796891464747	3796
consoles_games	138.49183817062476	1137
furniture_decor	87.56449364051431	8334
health_beauty	130.1635305067172	9670
fashion_shoes	89.93423664122166	262
computers_accessories	116.51390315574682	7827

Query result Insight 12

INTERPRETATION OF INSIGHT 12 OUTPUT

Olist can use this information to understand the performance and appeal of different categories within their market, which is valuable information. Olist can decide on appropriate pricing strategies and spot opportunities for aggressive pricing by analysing the average prices. Olist can determine the popularity and demand for each category by looking at the order counts, which enables them to allocate resources effectively and focus marketing efforts. For instance, the data shows that the perfumery category has a significant number of orders and a high average price, indicating that it is well-liked by customers. The fashion_shoes category, on the

other hand, has fewer orders and a lower average price, indicating potential areas for improvement. Olist will be able to increase sales, improve customer satisfaction, and keep a competitive edge in the e-commerce market by utilising this information to make data-driven decisions to optimise their product assortment, pricing strategies, and marketing initiatives.

MONGODB QUERIES AND INDEX

REFERENCE FORM

In order to generate the desired collections based on searches, aggregate queries like $lookup, $unwind, $project, and $out were used to convert the tables into documents. These queries were crucial in converting the relational data to a MongoDB structure based on documents. By using the specific identifiers (object_id) between tables, the $lookup stage was crucial in joining collections. In order to effectively replicate the relationships found in the original tables, this stage enabled the retrieval of relevant data from various collections.

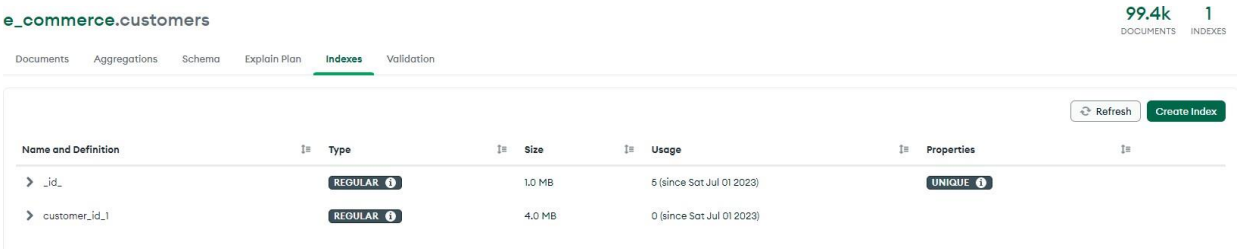
Additionally, embedded arrays were flattened using the $unwind stage, which also produced separate documents for each element of the array. This step made it possible to process and analyse the arrays' individual components in more detail. The document schemas were modified using the $project stage, which allowed users to choose which fields to include or exclude from the final collections. The customization of the resulting document structures based on the demands of the queries and the desired data representation was made easier by this stage.

The decision to forego using the embedded form was probably made as a result of conversion-related constraints. It may have led to unwieldy document sizes or complex data structures if all the tables had been included in a single document. When there are numerous possible combinations and when the particular objectives of the queries demand more flexibility in document design, embedding becomes less appropriate.

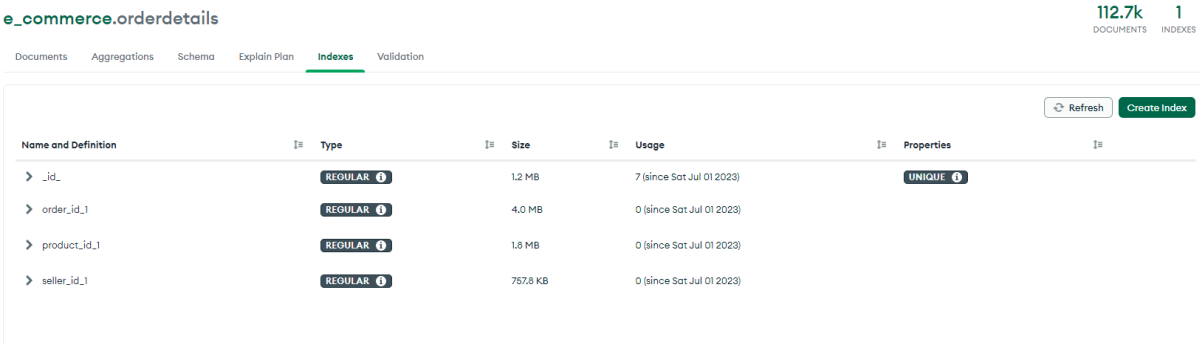
Additionally, in situations where data relationships are complicated and efficient querying is necessary, using the reference technique with distinct collections and indexes offers better scalability and performance. Independent updates to pertinent data may be made using the reference form without having an impact on the entire document. Additionally, it offers the ability to use MongoDB's referencing features to perform complex queries, aggregations, and joins.

In conclusion, the need for scalability, flexibility, and effective querying led us to choose the reference technique over the embedded form. The project produced a document-based database structure that better met the needs of the application by utilising aggregate queries and references. Below are the collections created and their specific indexes:

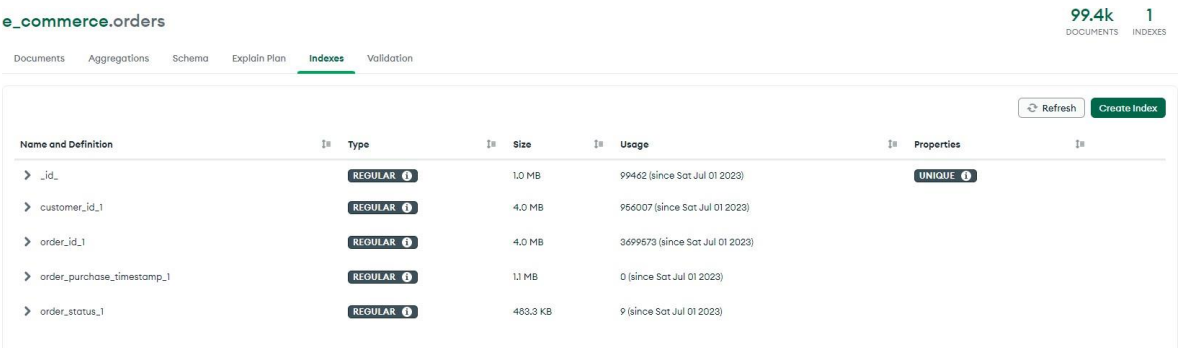
Indexes of Collection customers



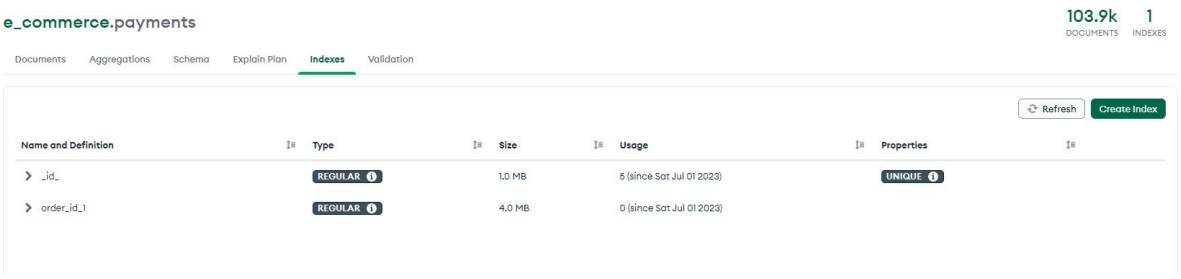
Indexes of Collection orderdetails



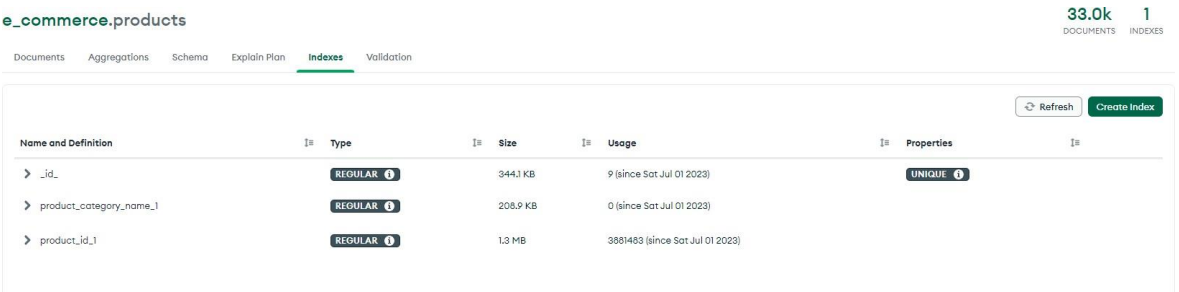
Indexes of Collection orders



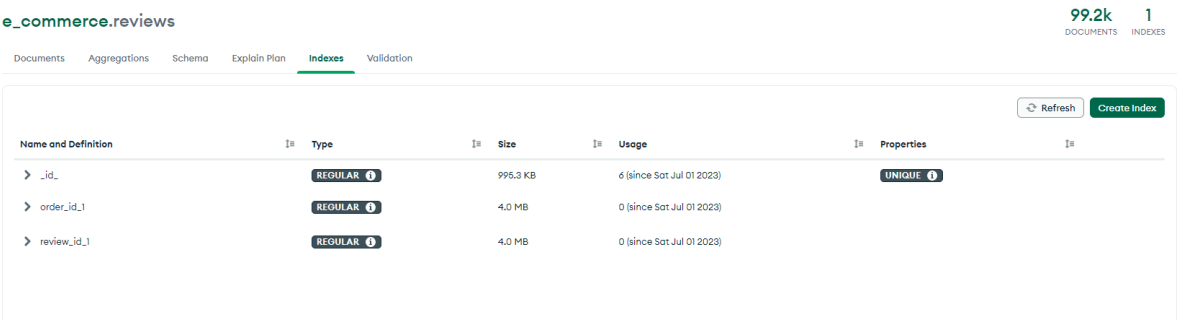
Indexes of Collection payments



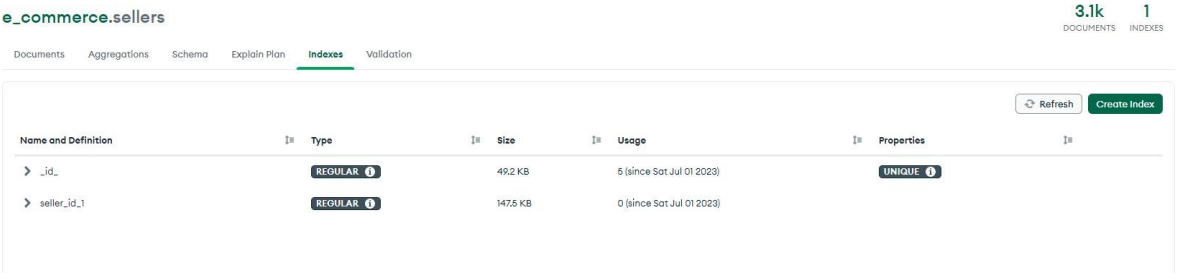
Indexes of Collection products



Indexes of Collection reviews



Indexes of Collection sellers



BUSINESS INSIGHT 1

MONGODB QUERY 1

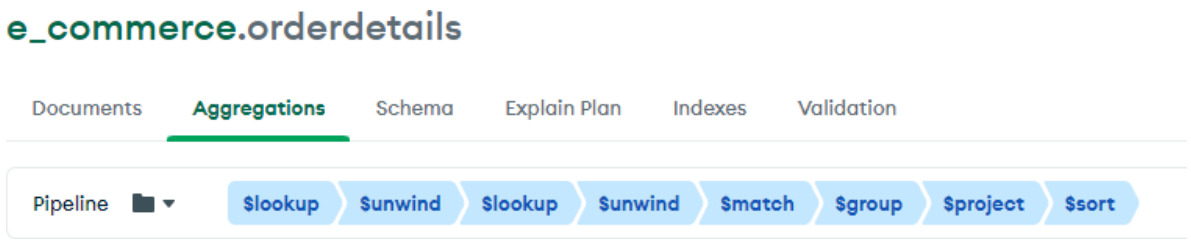
```
db.getCollection('customers').aggregate(
 [
  {
   $lookup: {
    from: 'orders',
    localField: 'customer_id',
    foreignField: 'customer_id',
    as: 'orders'
   }
  },
  {
   $unwind: {
    path: '$orders',
    includeArrayIndex: 'string',
    preserveNullAndEmptyArrays: true
   }
  },
  {
   $lookup: {
    from: 'payments',
    localField: 'orders.order_id',
    foreignField: 'order_id',
    as: 'payments'
   }
  },
```

```
  { $unwind: { path: '$payments' } },
  {
   $group: {
    _id: {
     customer_id: '$customer_id',
     customer_city: '$customer_city',
     order_id: '$orders.order_id'
    },
    Total_Purchase_Prices: {
     $sum: '$payments.payment_value'
    }
   }
  },
  {
   $project: {
    _id: 0,
    Customer_id: '$_id.customer_id',
    Customer_City: '$_id.customer_city',
    Total_Purchase_Prices: 1
   }
  },
  { $sort: { Total_Purchase_Prices: -1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```

```
Total_Purchase_Prices: 13664.08
Customer_id: "1617b1357756262bfa56ab541c47bc16"
Customer_City: "rio de janeiro"

Total_Purchase_Prices: 7274.88
Customer_id: "ec5b2ba62e574342386871631fafd3fc"
Customer_City: "vila velha"

Total_Purchase_Prices: 6929.31
Customer_id: "c6e2731c5b391845f6800c97401a43a9"
Customer_City: "campo grande"

Total_Purchase_Prices: 6922.21
Customer_id: "f48d464a0baaea338cb25f816991ab1f"
Customer_City: "vitoria"

Total_Purchase_Prices: 6726.66
Customer_id: "3fd6777bbce08a352fddd04e4a7cc8f6"
Customer_City: "marilia"

Total_Purchase_Prices: 6081.54
Customer_id: "05455dfa7cd02f13d132aa7a6a9729c6"
Customer_City: "divinopolis"
```


Insight 1 Output

BUSINESS INSIGHT 2



MONGODB QUERY 2

```
db.getCollection('orderdetails').aggregate(
 [
  {
   $lookup: {
    from: 'products',
    localField: 'product_id',
    foreignField: 'product_id',
```

```
    as: 'product_info'
   }
  },
  { $unwind: '$product_info' },
  {
   $lookup: {
    from: 'orders',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'order_info'
   }
  },
  { $unwind: '$order_info' },
  {
   $match: {
    'order_info.order_status': 'delivered'
   }
  },
  {
   $group: {
    _id: '$product_info.product_category_name',
    Total_sales: {
     $sum: {
      $add: ['$price', '$freight_value']
     }
    }
   }
  },
  {
   $project: {
    Product_Name: '$_id',
    Total_sales: 1,
    _id: 0
   }
```

```
    },
    { $sort: { Total_sales: -1 } }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```

Insight2 Output

BUSINESS INSIGHT 3

MONGODB QUERY 3

```
db.getCollection('orderdetails').aggregate(
  [
```

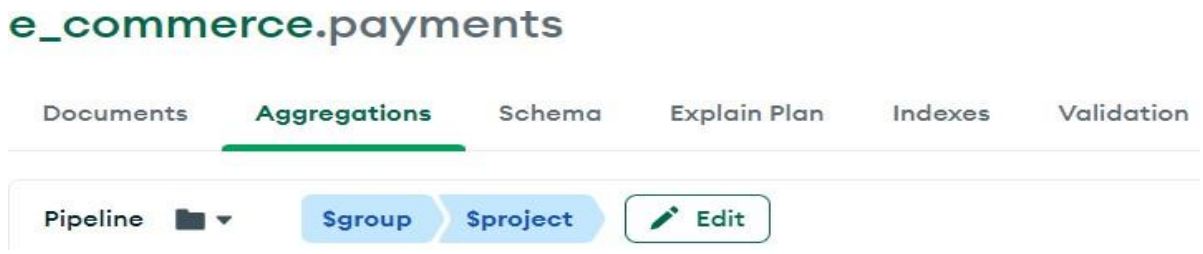
```
  {
   $lookup: {
    from: 'products',
    localField: 'product_id',
    foreignField: 'product_id',
    as: 'product'
   }
  },
  { $unwind: { path: '$product' } },
  {
   $group: {
    _id: {
     product_id: '$product.product_id',
     Product_Name:
     '$product.product_category_name'
    },
    Total_sales: {
     $sum: {
      $add: ['$price', '$freight_value']
     }
    }
   }
  },
  {
   $project: {
    Product_id: '$_id.product_id',
    Product_Name: '$_id.Product_Name',
    Total_sales: 1,
    _id: 0
   }
  },
  { $sort: { Total_sales: -1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
```

);


```
Total_sales: 67606.1
Product_id: "bb50f2e236e5eea0100680137654686c"
Product_Name: "health_beauty"

Total_sales: 60976.03
Product_id: "d1c427060a0f73f6b889a5c7c61f2ac4"
Product_Name: "computers_accessories"

Total_sales: 59093.99
Product_id: "6cdd53843498f92890544667809f1595"
Product_Name: "health_beauty"

Total_sales: 51071.6
Product_id: "99a4788cb24856965c36a24e339b6058"
Product_Name: "bed_bath_table"
```


Insight 3 Output

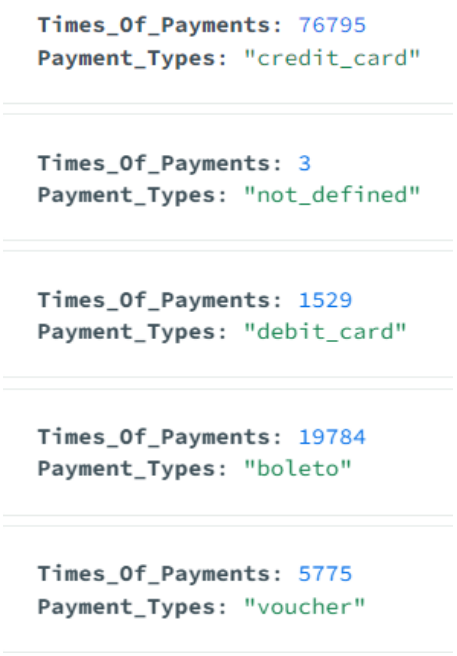
BUSINESS INSIGHT 4



MONGODB QUERY 4

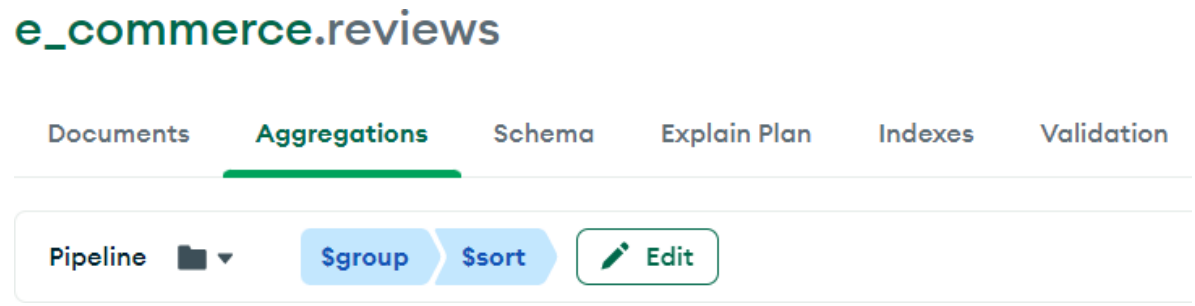
```
db.getCollection('payments').aggregate(
 [
  {
   $group: {
    _id: '$payment_type',
    Times_Of_Payments: { $sum: 1 }
   }
  },
  {
   $project: {
    Payment_Types: '$_id',
    Times_Of_Payments: 1,
```

```
      _id: 0
    }
   }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```



Insight 4 Output

BUSINESS INSIGHT 5



MONGODB QUERY 5

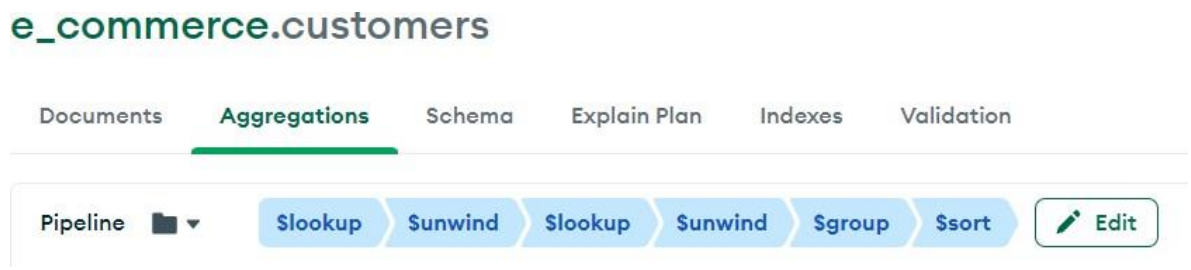
```
db.getCollection('reviews').aggregate(
 [
```

```
  {
    $group: {
      _id: '$review_score',
      count: { $sum: 1 }
    }
  },
  { $sort: { count: -1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```

```
_id: 5
count: 57328

_id: 4
count: 19142

_id: 1
count: 11424

_id: 3
count: 8179

_id: 2
count: 3151
```

Insight 5 Output

BUSINESS INSIGHT 5(2)



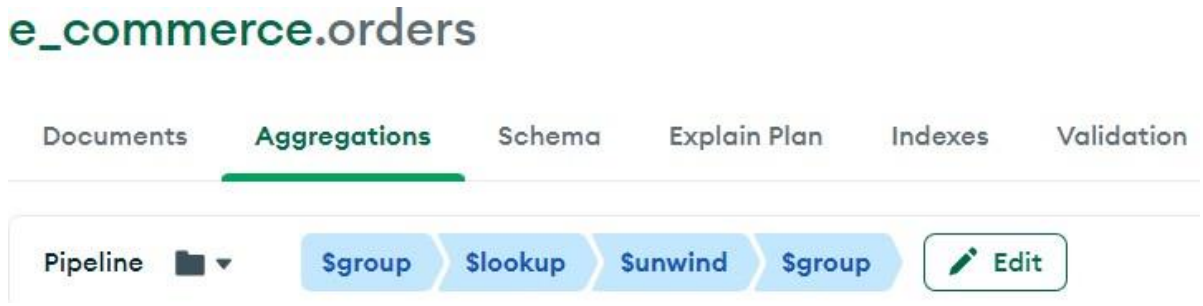
MONGODB QUERY 5(2)

```
db.getCollection('customers').aggregate(
 [
  {
   $lookup: {
    from: 'orders',
    localField: 'customer_id',
    foreignField: 'customer_id',
    as: 'order'
   }
  },
  { $unwind: { path: '$order' } },
  {
   $lookup: {
    from: 'reviews',
    localField: 'order.order_id',
    foreignField: 'order_id',
    as: 'reviews'
   }
  },
  { $unwind: { path: '$reviews' } },
  {
   $group: {
    _id: '$customer_state',
    average_score: {
     $avg: '$reviews.review_score'
```

```
        }
      }
    },
    { $sort: { average_score: -1 } }
  ],
  { maxTimeMS: 0, allowDiskUse: true }
);
```

Insight 5(2) Output

BUSINESS INSIGHT 5(3)



MONGODB QUERY 5(3)

```
db.getCollection('orders').aggregate(
 [
  {
   $group: {
    _id: '$customer_id',
    num_orders: { $sum: 1 }
   }
  },
  {
   $lookup: {
    from: 'orders',
    localField: '_id',
    foreignField: 'customer_id',
    as: 'order'
   }
  },
  { $unwind: { path: '$order' } },
  {
   $group: {
    _id: {
     $year: {
      $toDate:
       '$order.order_purchase_timestamp'
     }
    },
```

```
      num_orders: { $sum: '$num_orders' }
     }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```

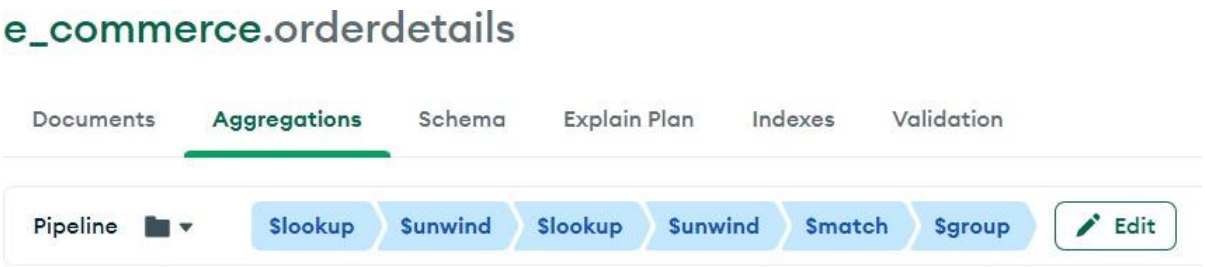
```
_id: 2018
num_orders: 54011
```

```
_id: 2016
num_orders: 329
```

```
_id: 2017
num_orders: 45101
```

Insight 5(3) Output

BUSINESS INSIGHT 5(4)



MONGODB QUERY 5(4)

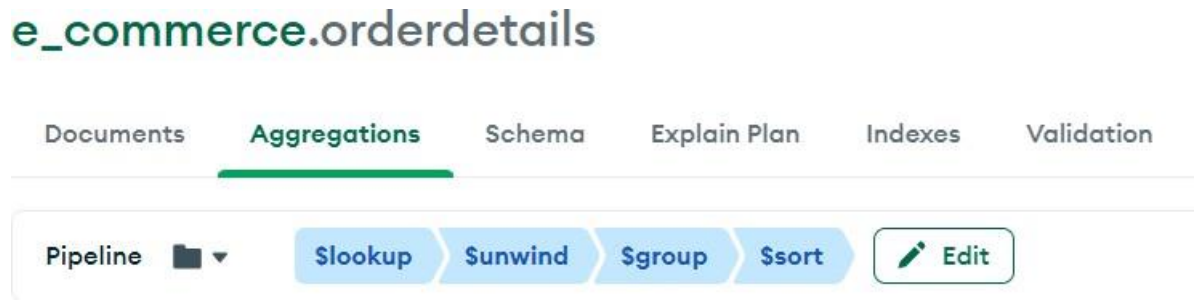
```
db.getCollection('orderdetails').aggregate(
  [
    {
     $lookup: {
       from: 'products',
       localField: 'product_id',
       foreignField: 'product_id',
       as: 'product'
```

```
      }
    },
    { $unwind: { path: '$product' } },
    {
     $lookup: {
       from: 'reviews',
       localField: 'order_id',
       foreignField: 'order_id',
       as: 'review'
     }
    },
    { $unwind: { path: '$review' } },
    { $match: { 'review.review_score': 5 } },
    {
     $group: {
       _id: '$product.product_id',
       product_category_name: {
         $first: '$product.product_category_name'
       },
       count: { $sum: 1 }
     }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```

```
_id: "8bd822c04ee6a8292a7b0debc3dbd74a"
product_category_name: "construction_tools_safety"
count: 1

_id: "38fbe42a76e0c841c9445467b7e5ab30"
product_category_name: "bed_bath_table"
count: 2

_id: "00878d953636afec00d3e85d55a12e7f"
product_category_name: "health_beauty"
count: 7

_id: "3918bb60ff48662029ca9f1e1f7b9842"
product_category_name: "fashion_bags_accessories"
count: 3

_id: "2d1e6a6907dcf4b2299622df6cc03c84"
product_category_name: "bed_bath_table"
count: 1
```


Insight 5(4) Output

BUSINESS INSIGHT 6



MONGODB QUERY 6

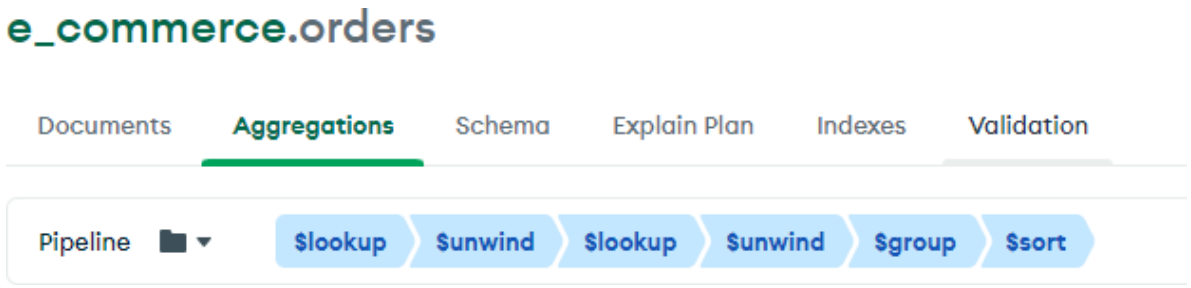
```
db.getCollection('orderdetails').aggregate(
 [
  {
   $lookup: {
    from: 'sellers',
    localField: 'seller_id',
    foreignField: 'seller_id',
    as: 'seller'
```

```
    }
  },
  { $unwind: { path: '$seller' } },
  {
   $group: {
    _id: '$seller.seller_id',
    order_count: { $sum: 1 },
    total_revenue: { $sum: '$price' }
   }
  },
  { $sort: { total_revenue: -1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```


```
_id: "4869f7a5dfa277a7dca6462dcf3b52b2"
order_count: 1156
total_revenue: 229472.63

_id: "53243585a1d6dc2643021fd1853d8905"
order_count: 410
total_revenue: 222776.05

_id: "4a3ca9315b744ce9f8e9374361493884"
order_count: 1987
total_revenue: 200472.92

_id: "fa1c13f2614d7b5c4749cbc52fecda94"
order_count: 586
total_revenue: 194042.03

_id: "7c67e1448b00f6e969d365cea6b010ab"
order_count: 1364
total_revenue: 187923.89

_id: "7e93a43ef30c4f03f38b393420bc753a"
order_count: 340
total_revenue: 176431.87
```


Insight 6 Output

BUSINESS INSIGHT 7



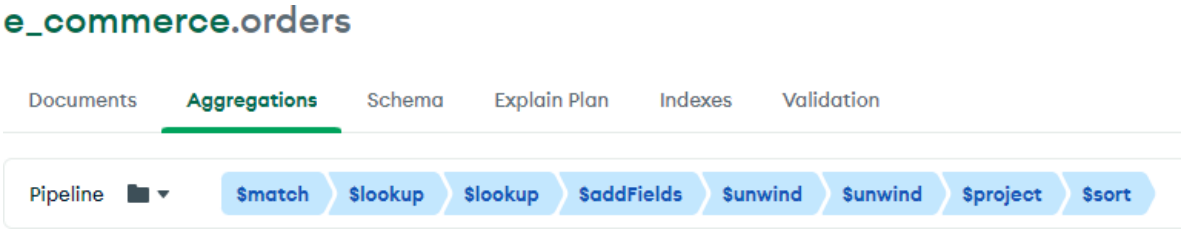
MONGODB QUERY 7

```
db.getCollection('orders').aggregate(
 [
  {
   $lookup: {
    from: 'customers',
    localField: 'customer_id',
    foreignField: 'customer_id',
    as: 'customer_info'
   }
  },
  { $unwind: '$customer_info' },
  {
   $lookup: {
    from: 'orderdetails',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'orderdetail_info'
   }
  },
  { $unwind: '$orderdetail_info' },
  {
   $group: {
    _id: '$customer_info.customer_state',
    customer_count: { $sum: 1 },
```

```
        total_sales: {
          $sum: '$orderdetail_info.price'
        }
      }
    },
    { $sort: { total_sales: -1 } }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```

Insight7 Output

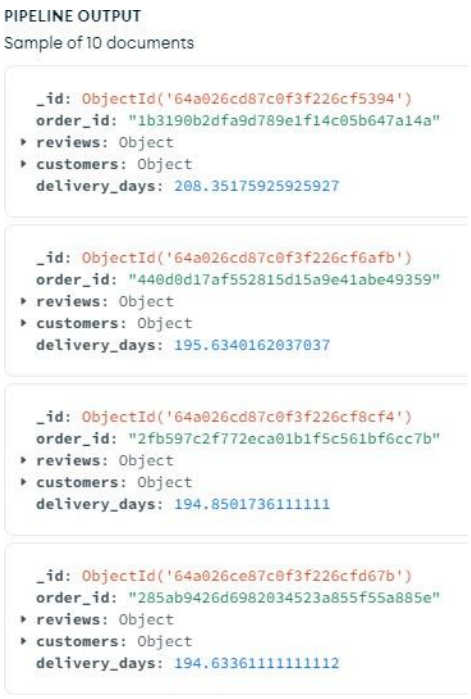
BUSINESS INSIGHT 8



MONGODB QUERY 8

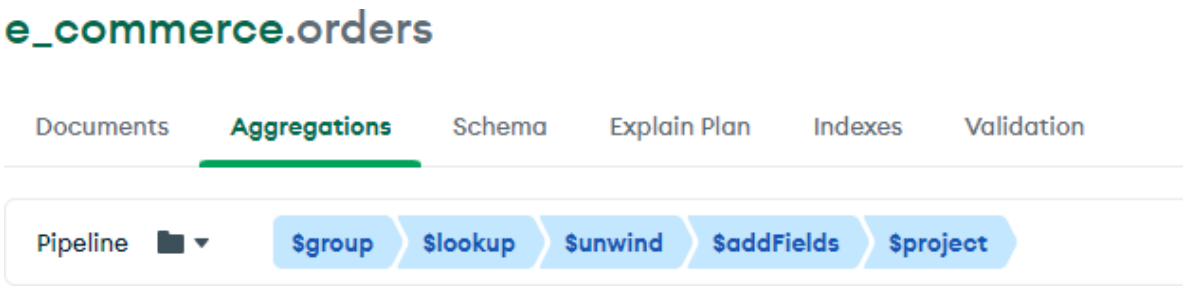
```
db.getCollection('orders').aggregate(
 [
  {
   $match: {
    order_status: 'delivered',
    order_delivered_customer_date: {
     $ne: null
    }
   }
  },
  {
   $lookup: {
    from: 'reviews',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'reviews'
   }
  },
  {
   $lookup: {
    from: 'customers',
    localField: 'customer_id',
    foreignField: 'customer_id',
    as: 'customers'
   }
```

```
    },
    {
     $addFields: {
       delivery_days: {
         $subtract: [
           {
             $toDate:
               '$order_delivered_customer_date'
           },
           {
             $toDate: '$order_purchase_timestamp'
           }
         ]
       }
     }
    },
    { $unwind: '$reviews' },
    { $unwind: '$customers' },
    {
     $project: {
       order_id: 1,
       'customers.customer_city': 1,
       'customers.customer_state': 1,
       delivery_days: {
         $divide: ['$delivery_days', 86400000]
       },
       'reviews.review_score': 1
     }
    },
    { $sort: { delivery_days: -1 } }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```



Insight8 Output

BUSINESS INSIGHT 9



MONGODB QUERY 9

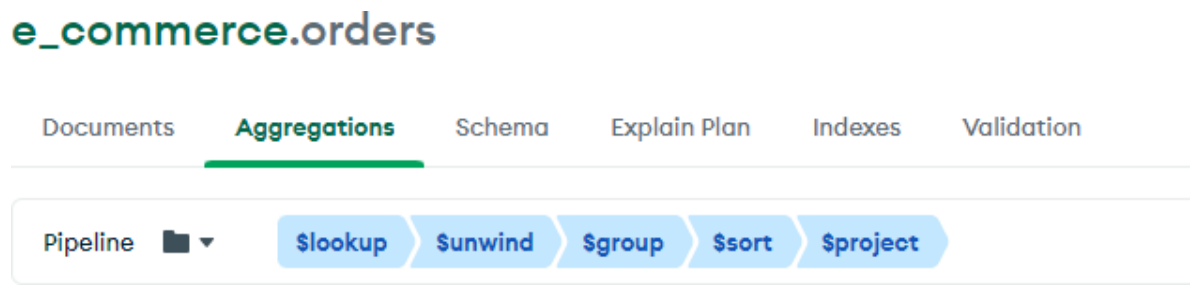
```
db.getCollection('orders').aggregate(
 [
  {
   $group: {
    _id: '$order_status',
    order_count: { $sum: 1 }
   }
  },
```

```
{
 $lookup: {
  from: 'orders',
  pipeline: [
   {
    $group: {
     _id: null,
     total: { $sum: 1 }
    }
   }
  ],
  as: 'totalOrders'
 }
},
{ $unwind: '$totalOrders' },
{
 $addFields: {
  percentage: {
   $multiply: [
    {
     $divide: [
      '$order_count',
      '$totalOrders.total'
     ]
    },
    100
   ]
  }
 }
},
{
 $project: {
  order_status: '$_id',
  order_count: 1,
```

```
        percentage: { $round: ['$percentage', 2] }
      }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```

Insight9 Output

BUSINESS INSIGHT 9(2)



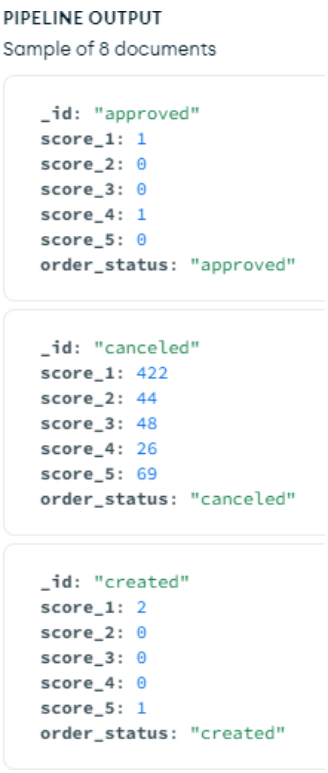
MONGODB QUERY 9(2)

```
db.getCollection('orders').aggregate(
 [
  {
   $lookup: {
    from: 'reviews',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'review_info'
   }
  },
  { $unwind: '$review_info' },
  {
   $group: {
    _id: '$order_status',
    score_1: {
     $sum: {
      $cond: [
       {
        $eq: [
         '$review_info.review_score',
         1
        ]
       },
       1,
```

```
        0
      ]
    }
  },
  score_2: {
    $sum: {
      $cond: [
        {
          $eq: [
            '$review_info.review_score',
            2
          ]
        },
        1,
        0
      ]
    }
  },
  score_3: {
    $sum: {
      $cond: [
        {
          $eq: [
            '$review_info.review_score',
            3
          ]
        },
        1,
        0
      ]
    }
  },
  score_4: {
    $sum: {
```

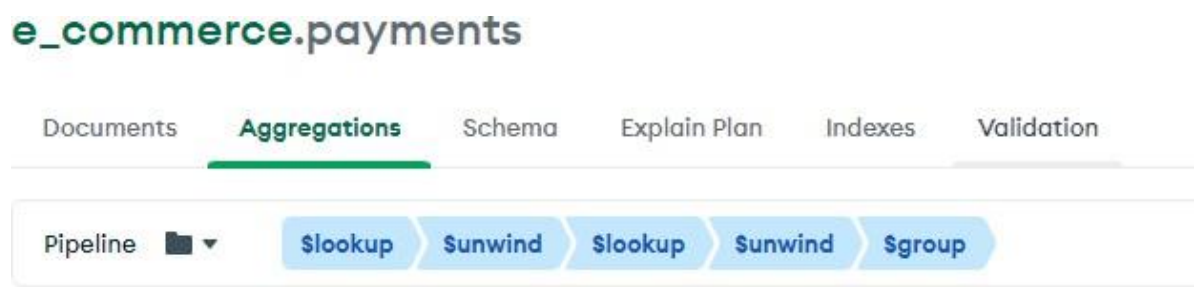
```
        $cond: [
          {
            $eq: [
              '$review_info.review_score',
              4
            ]
          },
          1,
          0
        ]
      }
    },
    score_5: {
      $sum: {
        $cond: [
          {
            $eq: [
              '$review_info.review_score',
              5
            ]
          },
          1,
          0
        ]
      }
    }
  }
},
{ $sort: { _id: 1 } },
{
  $project: {
    order_status: '$_id',
    score_1: 1,
    score_2: 1,
```

```
        score_3: 1,
        score_4: 1,
        score_5: 1
      }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```



Insight9(2) Output

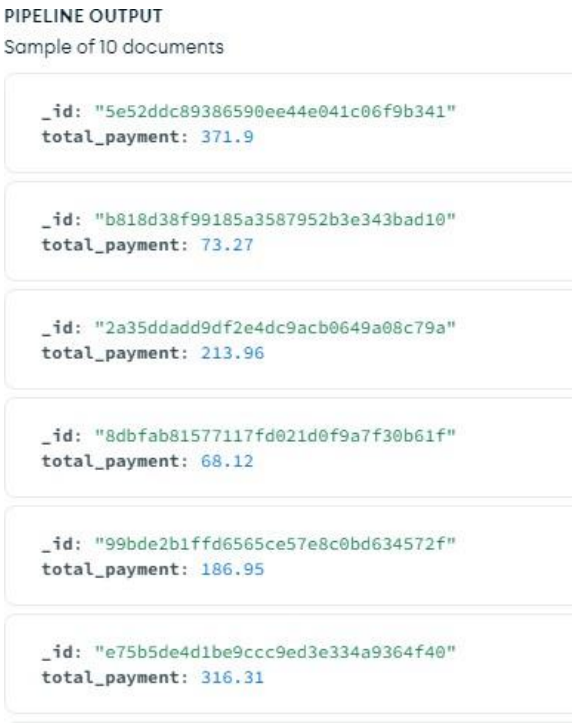
BUSINESS INSIGHT 10



MONGODB QUERY 10

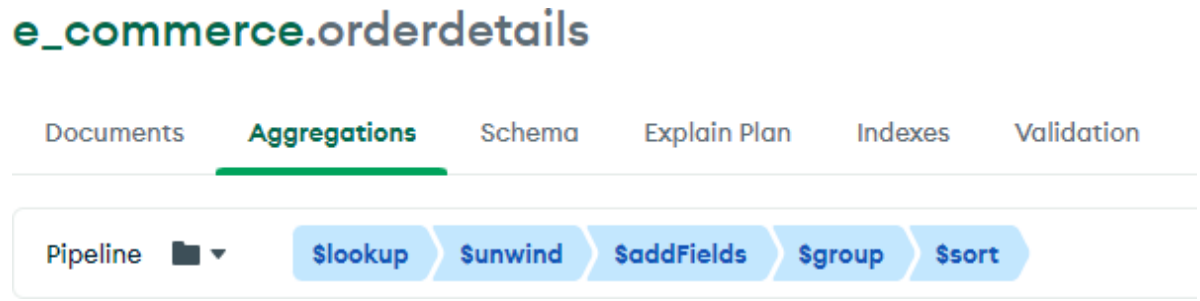
```
db.getCollection('payments').aggregate(
 [
  {
   $lookup: {
    from: 'orders',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'order_info'
   }
  },
  { $unwind: '$order_info' },
  {
   $lookup: {
    from: 'customers',
    localField: 'order_info.customer_id',
    foreignField: 'customer_id',
    as: 'customer_info'
   }
  },
  { $unwind: '$customer_info' },
  {
   $group: {
    _id: '$customer_info.customer_id',
    total_payment: { $sum: '$payment_value' }
```

```
      }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);
```



Insight10 Output

BUSINESS INSIGHT 11



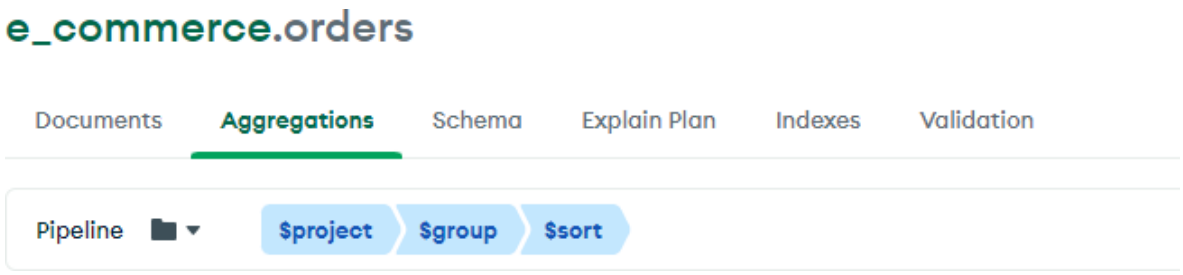
MONGODB QUERY 11

```
db.getCollection('orderdetails').aggregate(
 [
  {
   $lookup: {
    from: 'orders',
    localField: 'order_id',
    foreignField: 'order_id',
    as: 'order_info'
   }
  },
  { $unwind: '$order_info' },
  {
   $addFields: {
    order_month: {
     $month:
      '$order_info.order_purchase_timestamp'
    },
    order_year: {
     $year:
      '$order_info.order_purchase_timestamp'
    }
   }
  },
  {
```

```
    $group: {
      _id: {
        month: '$order_month',
        year: '$order_year'
      },
      total_sales: {
        $sum: {
          $multiply: [
            '$price',
            '$order_item_id'
          ]
        }
      }
    }
  },
  { $sort: { '_id.year': 1, '_id.month': 1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```

Insight11 Output

BUSINESS INSIGHT 11(2)



MONGODB QUERY 11(2)

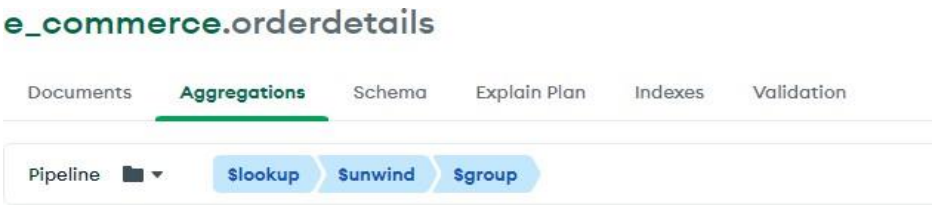
```
db.getCollection('orders').aggregate(
 [
  {
   $project: {
    time_of_day: {
     $let: {
      vars: {
       hourOfDay: {
        $hour: '$order_purchase_timestamp'
       }
      },
      in: {
       $cond: [
        { $lte: ['$$hourOfDay', 5] },
        'Dawn',
        {
         $cond: [
          {
           $and: [
            {
             $gte: ['$$hourOfDay', 6]
            },
            {
             $lte: [
```

```
                '$$hourOfDay',
                11
              ]
            }
          ]
        },
        'Morning',
        {
          $cond: [
            {
              $and: [
                {
                  $gte: [
                    '$$hourOfDay',
                    12
                  ]
                },
                {
                  $lte: [
                    '$$hourOfDay',
                    17
                  ]
                }
              ]
            },
            'Afternoon',
            'Night'
          ]
        }
      ]
    }
  ]
 }
}
```

```
      }
    }
  },
  {
    $group: {
      _id: '$time_of_day',
      num_orders: { $sum: 1 }
    }
  },
  { $sort: { num_orders: -1 } }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```

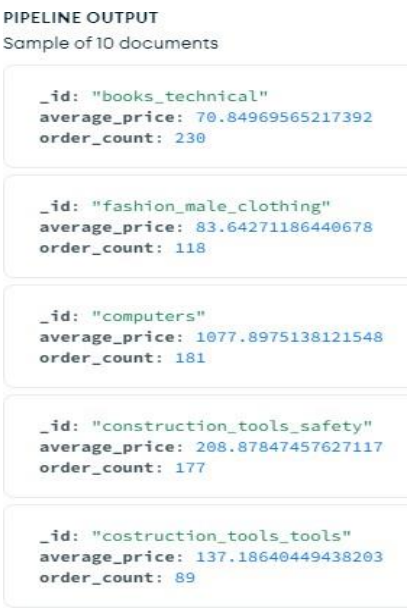
Insight11(2) Output

BUSINESS INSIGHT 12



MONGODB QUERY 12

```
db.getCollection('orderdetails').aggregate(
 [
  {
   $lookup: {
    from: 'products',
    localField: 'product_id',
    foreignField: 'product_id',
    as: 'product_info'
   }
  },
  { $unwind: '$product_info' },
  {
   $group: {
    _id: '$product_info.product_category_name',
    average_price: { $avg: '$price' },
    order_count: { $sum: 1 }
   }
  }
 ],
 { maxTimeMS: 60000, allowDiskUse: true }
);
```



Insight12 Output

SQL vs NoSQL

The document-based data model is more applicable and suitable for the Olist dataset project. The justifications are as follows:

1. Flexibility and Scalability: The Olist dataset contains information about customers, orders, products, and reviews from a variety of sources. The diverse and dynamic nature of the data can be handled with a document-based data model like MongoDB. It permits a flexible schema that can accommodate the various attributes and relationships found in the dataset by allowing each document to have a different structure. This adaptability makes it simpler to manage updates and new dataset additions without needing to change the database schema.

2. Complex Relationships: The Olist dataset contains relationships that are intricate between different types of entities, including orders, clients, merchandise, and reviews. Using document-based databases, we can represent one-to-many and even many-to-many relationships by embedding related data in a single document. By eliminating the need for joins between multiple tables, this embedded approach makes

querying easier and boosts performance. As related information is stored together, it also ensures data locality and reduces the need for separate database lookups.

3. Performance and Scalability: Horizontal scalability and performance are areas where document-based databases shine. They can effectively handle large volumes of read and write operations and distribute data across multiple servers. The document-based approach can offer better performance and scalability in the case of Olist, which deals with a large dataset and potentially high traffic, than a conventional relational database.

4. Document-based data models allow for agile development because they make it simple to iterate and modify database structures as application requirements change. The quick iterations and lack of strict schema migrations common to relational databases are made possible by the flexible schema.

5. Ecosystem and Tooling: Document-based databases have a developed ecosystem with a wide selection of tools, libraries, and community support, such as MongoDB. Powerful features like flexible querying, indexing choices, and seamless integration with well-liked frameworks and languages are made available to developers by this ecosystem.

Due to its flexibility, performance, scalability, and ease of development, the document-based data model is an excellent choice for the Olist case study due to the dataset's complexity and diversity.

CONCLUSION

In conclusion, the data and queries run have been a huge asset to Olist, an e-commerce platform. By carefully taking into account various factors like sales patterns, customer behaviour, payment methods, and product reviews, Olist can optimise its operations, increase customer satisfaction, and foster business growth. Olist uses the data to better understand long-term sales trends, identify peak times, and allocate resources. It can be used by Olist to create customised marketing campaigns, identify popular product categories, and better

manage inventory. Olist also makes use of customer review analysis to address issues, improve the quality of its products, and improve customer service.

Additionally, the performance of queries can be greatly enhanced by adding indexes to the database tables. Indexes help to speed up data retrieval by allowing the database engine to easily find the required data based on the indexed columns. This decreases waiting time and improves the efficiency of query execution. By correctly indexing the relevant columns used in the queries, Olist can experience faster query response times, lower CPU and disc usage, and improved overall database performance. Because indexes can expedite the quick retrieval of specific data subsets, this is especially helpful when working with large amounts of data.

NoSQL databases, like MongoDB, have a number of benefits over conventional SQL databases when handling Olist data. The flexible schema design offered by NoSQL databases enables the use of dynamic and changing data structures. They are made to be horizontally scaleable, which is essential for effectively handling large amounts of data and high read/write loads. Additionally, NoSQL databases provide strong frameworks for aggregation that can manage intricate relationships and aggregations between collections. NoSQL databases' document-oriented storage is in line with the semi-structured and unstructured data that is typically present in e-commerce platforms. Furthermore, NoSQL databases enable effective indexing, which can greatly enhance query performance and cut down on waiting time, especially when working with big datasets.

Olist can benefit from the scalability, robust aggregation capabilities, document-oriented storage, and scalable indexing offered by NoSQL databases like MongoDB by utilising them. These tools help Olist manage and analyse its data effectively, make data-driven decisions, increase operational effectiveness, improve customer experiences, and maintain its competitiveness in the fast-paced e-commerce market.