



AMAZON BRAZIL'S DATA ANALYSIS

- To analyse Amazon Brazil's data to identify trends, customer behaviours, and preferences that could be leveraged in the Indian market

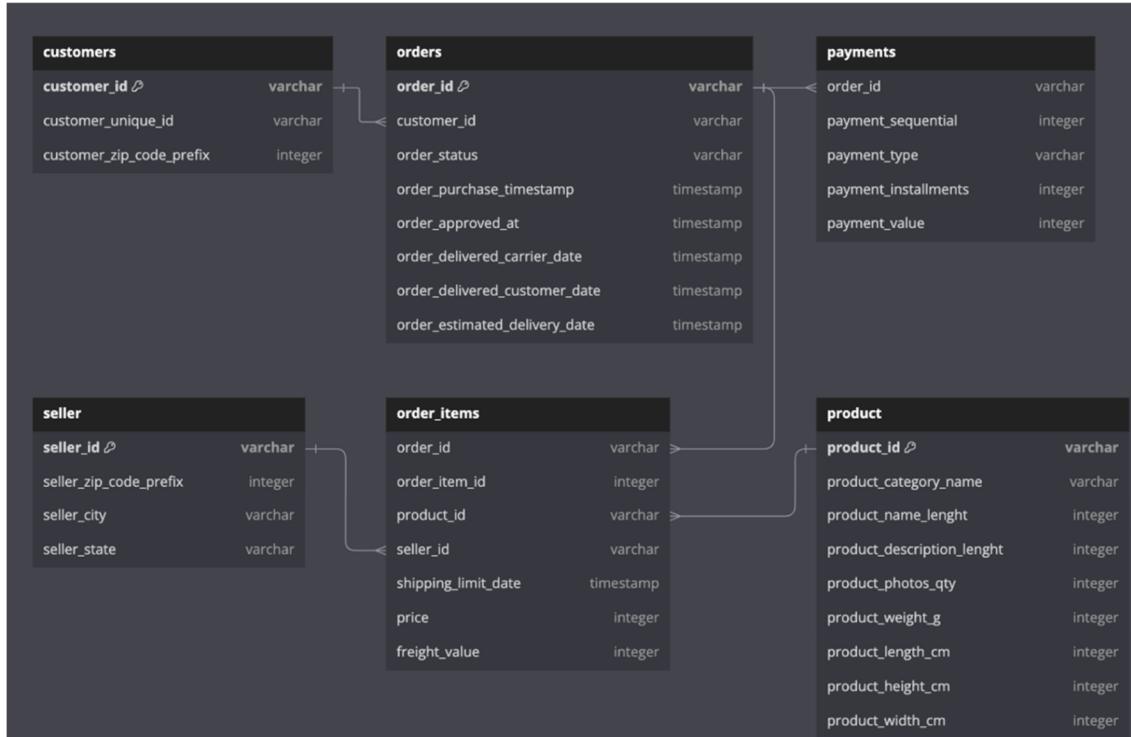
-By Moganapriyaa V

Introduction

[Amazon](#), a global leader in e-commerce, has achieved significant success in markets like the U.S., Europe, and Asia. In Brazil, Amazon connects small and medium businesses with millions of customers, becoming a key player. Given the similarities between Brazil and India—such as large populations and diverse consumer bases—there's an opportunity to replicate success in India.

Overview of the Schema

Below is the schema of the provided data, **including relationships and primary keys** for each table



- Customers:** Holds customer information, including unique identifiers and locations, to study customer demographics and behavior.
- Orders:** Captures details about each order, such as order status and timestamps, essential for tracking the order lifecycle.
- Order Items:** Lists each item in an order with details like price, seller, and shipping information.
- Product:** Contains product specifications, such as category, size, and weight, for product-level analysis.

- **Seller:** Provides data about sellers, including their location and unique identifiers, to evaluate seller performance.
- **Payments:** Records transaction details, such as payment type and value, to understand payment preferences and financial performance.

Analysis 1.1

Problem Statement

To simplify its financial reports, Amazon India needs to standardize payment values. Round the average payment values to integer (no decimal) for each payment type and display the results sorted in ascending order.

Table Used : Amazon_Brazil.Payments

Columns Used : Payment Type, Payment Value

Functions Used :

- **Round ()** – To round the payment value without decimal
- **Avg()** – To find the average payment value
- **Group by ()** – To consolidate the payment values payment type
- **Order by ()** – To arrange values in descending order

SQL Query -

```
Select Payment_Type, Round(Avg(Payment_value)) as rounded_avg_payment
from Amazon_Brazil.Payments
where Payment_Type <> 'not_defined'
group by Payment_Type
order by rounded_avg_payment desc
```

Results -

	payment_type character varying (200)	rounded_avg_payment double precision
1	credit_card	163
2	boleto	145
3	debit_card	143
4	voucher	66

Analysis 1.2

Problem Statement

To refine its payment strategy, Amazon India wants to know the distribution of orders by payment type. Calculate the percentage of total orders for each payment type, rounded to one decimal place, and display them in descending order.

Table Used : Amazon_Brazil.Payments

Columns Used : Payment Type, Order_id

Functions Used :

- **Round ()** – To round the payment value without decimal
- **Count ()** – To count the total number of order id
- **Group by ()** – To consolidate the payment values payment type
- **Order by ()** – To arrange values in descending order

SQL Query -

```
select payment_type,  
round(count(order_id) * 100.0/(select count(*) from amazon_brazil.payments),1)  
as percentage_orders  
from amazon_brazil.payments  
group by payment_type  
order by percentage_orders desc;
```

Results -

	payment_type character varying (200)	percentage_orders numeric
1	credit_card	73.9
2	boleto	19.0
3	voucher	5.6
4	debit_card	1.5

Recommendations for Analysis 1.1 and 1.2 based on Payment methods

1. Credit Card – 73.9% (Dominant method)

The majority of customers already use credit cards, indicating trust and ease of use. This is a strong channel for upselling and increasing purchase frequency.

- **Promote EMI / Installment plans**
- **Credit card–exclusive discounts & cashback**
- **Push loyalty rewards**

2. Boleto – 19% (Key method for non-credit card population)

Boleto is widely used by price-sensitive or unbanked customers. It is popular in markets like Brazil for customers without credit history.

- **Offer extended payment deadline options**
- **Send automated reminders** before Boleto expires

3. Voucher – 5.6% (Useful but underutilized)

Voucher usage is low. This method is often associated with promotions, refunds, gift cards, and employer benefit programs.

- **Strengthen marketing of gift cards** before festive seasons.
- **Offer vouchers as compensation** for delayed deliveries (keeps money inside the ecosystem)

4. Debit Card – 1.5% (Very low adoption)

Debit card use is very low

- **Simplify authentication flow** (e.g., saved cards, one-tap payments where possible).
- **Promote debit card cashback/instant discounts** to make it appealing

Analysis 1.3

Problem Statement

Amazon India seeks to create targeted promotions for products within specific price ranges. Identify all products priced between 100 and 500 BRL that contain the word 'Smart' in their name. Display these products, sorted by price in descending order.

Table Used : Amazon_Brazil.Products, Amazon_Brazil.Order_Items

Columns Used : Product_id, Product_Category_Name

Functions Used :

- **Join ()** – To combine the tables Amazon_Brazil.Products and Amazon_Brazil.Order_Items
- **Like ()** – To filter out product having smart in the product category name
- **Between ()** – To find values between the range
- **Order by ()** – To arrange values in descending order

SQL Query -

```
Select O.product_id, Price, Product_category_name  
from Amazon_Brazil.order_items O  
JOIN Amazon_Brazil.Products P on P.product_id = O.product_id  
where price between 100 and 500 and Product_category_name like ('%smart%')  
ORDER BY Price desc;
```

Results -

Query returned 34 rows, here only 15 rows are shown

	product_id character varying (200)	price double precision	product_category_name character varying
1	1df1a2df8ad2b9d3aa49fd851e314...	439.99	smart
2	7debe59b10825e89c1cbcc8b190c...	349.99	smart
3	ca86b9fe16e12de698c955aedff0ae...	349	smart
4	ca86b9fe16e12de698c955aedff0ae...	349	smart
5	0e52955ca8143bd179b311cc454a...	335	smart
6	7aeaa8f3e592e380c420e8910a717...	329.9	smart
7	7aeaa8f3e592e380c420e8910a717...	329.9	smart
8	7aeaa8f3e592e380c420e8910a717...	329.9	smart
9	7aeaa8f3e592e380c420e8910a717...	329.9	smart
10	7aeaa8f3e592e380c420e8910a717...	329.9	smart
11	7aeaa8f3e592e380c420e8910a717...	329.9	smart
12	d1b571cd58267d8cac8b2af6e288...	299.9	smart
13	d1b571cd58267d8cac8b2af6e288...	299.9	smart
14	66ffe28d0fd53808d0535eee4b90a...	254	smart
15	f06796447de379a26dde5fcac6a1a...	239.9	smart

Recommendations

- Provide limited-time price drops (5–15%) to create urgency which leads to more sales.
- Run **sponsored product ads** for items priced ₹300–₹400 so that increasing the sales

Analysis 1.4

Problem Statement

To identify seasonal sales patterns, Amazon India needs to focus on the most successful months. Determine the top 3 months with the highest total sales value, rounded to the nearest integer.

Table Used : Amazon_Brazil.Orders, Amazon_Brazil.Orders_items

Columns Used : Price, Order_purchase_Timestamp

Functions Used :

- Extract ()** – To get month from a date
- Sum ()** – To finsum the price of a particular value
- Group by ()** – To consolidate the sales basedon month
- Join()** – To combine Amazon_Brazil.Orders and Amazon_Brazil.Orders_items table
- Order by ()** – To arrange values in descending order
- Limit()** – To get the top 3 months

SQL Query -

```
select extract(month from o.order_purchase_timestamp) as month,  
round(sum(i.price)) as total_sales  
from amazon_brazil.orders o  
join amazon_brazil.order_items i  
on o.order_id = i.order_id  
group by month  
order by total_sales desc  
limit 3
```

Results -

	month numeric	total_sales double precision
1	5	1502589
2	8	1428658
3	7	1393539

Recommendations

- May generates the highest total sales. Launching major seasonal offers, and marketing campaigns during this month can help to further boost revenue.
- July and August are also strong sales months. Implementing targeted sales strategies such as end-of-season sales, and new product launch can help to further strengthen the sales figures.

Analysis 1.5

Problem Statement

Amazon India is interested in product categories with significant price variations. Find categories where the difference between the maximum and minimum product prices is greater than 500 BRL.

Table Used : Amazon_Brazil.Products, Amazon_Brazil.Orders_items

Columns Used : Product_Category_Name, Price, Product_id

Functions Used :

- **Min () &Max()** – To find min and max price value
- **Join()** – To combine Amazon_Brazil.Products and Amazon_Brazil.Orders_items table
- **Group by ()** – To consolidate the price based on Product_Category_name
- **Order by ()** – To arrange values in descending order

SQL Query -

```
Select P.product_category_name, Round(Max(price) - Min(price)) as Price_difference from
Amazon_Brazil.products P
JOIN Amazon_Brazil.order_items i
on P.product_id = i.product_id
Group by P.product_category_name
having (MAX(price) - MIN(price)) > 500;
```

Results -

Sql Query returned 57 rows. 10 rows are shown here.

	product_category_name character varying	price_difference double precision
1	climatizacao	1588.1
2	livros_importados	730.01
3	[null]	3977
4	ferramentas_jardim	3923.65
5	dvds_blu_ray	1411.1
6	cine_foto	867.19
7	beleza_saude	3122.8
8	livros_interesse_geral	893.9
9	tablets_impressao_imagem	875.09
10	papelaria	1690.71

Recommendations

- For High variation categories like **consoles_games, ferramentas_jardim , bebes, beleza_saude, cama_mesa_banho, climatizacao** - offering seasonal discounts promotes sales
- For medium variation categories like **cine_foto ,tablets_impressao_imagem, livros_interesse_geral** - Add small discounts (5–7%), Highlighting free delivery or fast shipping

Analysis 1.6

Problem Statement

To enhance the customer experience, Amazon India wants to find which payment types have the most consistent transaction amounts. Identify the payment types with the least variance in transaction amounts, sorting by the smallest standard deviation first.

Table Used : Amazon_Brazil.Payments

Columns Used : Product_Type, Payment_Value

Functions Used :

- **STDDEV_SAMP** – To find the standard deviation of payments
- **Group by ()** – To consolidate the payments based on Payment type
- **Order by ()** – To arrange values in ascending order

SQL Query -

```
SELECT payment_type, Round(STDDEV_SAMP(payment_value) )  
AS std_deviation  
FROM Amazon_Brazil.payments  
Where payment_type <> 'not_defined'  
group by payment_type  
order by std_deviation
```

Results -

	payment_type character varying (200)	std_deviation double precision
1	debit_card	246
2	credit_card	222
3	boleto	214
4	voucher	116

Recommendations

1. Promote voucher payments:

- Voucher payments have the least standard deviation (116). Encouraging customers to use vouchers can help streamline payment processes and predict transaction patterns more accurately.

2. Improving boleto and credit card support:

- Both boleto and credit card payments have moderate deviation (214 and 222, respectively). Ensure robust support for these payment methods, as they provide predictable transaction amounts.

Analysis 1.7

Problem Statement

Amazon India wants to identify products that may have incomplete name in order to fix it from their end. Retrieve the list of products where the product category name is missing or contains only a single character.

Table Used : Amazon_Brazil.Products

Columns Used : Product_id, Product_category_name

Functions Used :

- **Null** – To find missing product_category_name
- **length ()** – To find the length of the product_category_name

SQL Query -

```
select product_id, product_category_name  
from amazon_brazil.products  
where product_category_name is null  
or length(Trim(product_category_name)) = 1;
```

Results –

Sql query returned 614 rows. Here 10 rows are displayed

	product_id [PK] character varying (200)	product_category_name character varying
1	a41e356c76fab66334f36de622ecbd...	[null]
2	d8dee61c2034d6d075997acef1870...	[null]
3	56139431d72cd51f19eb9f7dae4d16...	[null]
4	46b48281eb6d663ced748f324108c...	[null]
5	5fb61f482620cb672f5e586bb132ea...	[null]
6	e10758160da97891c2fdcbc35f0f03...	[null]
7	39e3b9b12cd0bf8ee681bbc1c130fe...	[null]
8	794de06c32a626a5692ff50e4985d3...	[null]
9	7af3e2da474486a3519b0cba9dea8...	[null]
10	629beb8e7317703dcc5f35b5463fd2...	[null]

Analysis 2.1

Problem Statement

Amazon India wants to understand which payment types are most popular across different order value segments (e.g., low, medium, high). Segment order values into three ranges: orders less than 200 BRL, between 200 and 1000 BRL, and over 1000 BRL. Calculate the count of each payment type within these ranges and display the results in descending order of count

Table Used : Amazon_Brazil.Payments

Columns Used : Product_Type, Payment_Value

Functions Used :

- **Case ()** – To segment values based on particular values
- **Group by ()** – To consolidate the payments based on order value segment
- **Order by ()** – To arrange values in descending order

SQL Query -

```
Select payment_type, count(payment_type)
case
when payment_value > 1000 then 'high'
when payment_value between 200 and 1000 then 'medium'
when payment_value < 200 then 'low'
else 'NA'
end as order_value_segment
from amazon_brazil.payments
group by order_value_segment, payment_type
order by count desc;
```

Results -

	payment_type character varying (200) 	count bigint 	order_value_segment text 
1	credit_card	60548	low
2	boleto	16444	low
3	credit_card	15303	medium
4	voucher	5476	low
5	boleto	3162	medium
6	debit_card	1287	low
7	credit_card	944	high
8	voucher	286	medium
9	debit_card	227	medium
10	boleto	178	high
11	debit_card	15	high
12	voucher	13	high

Recommendations

Prioritize popular payment methods:

- Credit cards are the most popular payment method across all order value segments, particularly in both low and medium-value orders. Offering rewards or discounts, could drive more sales and enhance user experience for credit card transactions.
- Boleto is the second most popular payment method. Promoting boleto payment options for higher-order values may encourage broader adoption.

Promote unpopular payment methods:

- Voucher usage is more common in low order value segments. Amazon could run promotions offering vouchers to increase sales in the medium and high segments.
- Debit card payments are less common. Amazon might consider offering additional incentives, such as cashback or instant rewards, to promote debit card usage across all order value segments.

Analysis 2.2

Problem Statement

Amazon India wants to analyse the price range and average price for each product category. Calculate the minimum, maximum, and average price for each category, and list them in descending order by the average price.

Table Used : Amazon_Brazil.Products, Amazon_Brazil.Order_items

Columns Used : Product_Type, Price

Functions Used :

- **Min () ,Max(),**– To get Minimum and maximum price values
- **Avg ()** – To get the average price values
- **Round ()** – to round the value without decimal
- **Group by ()** – To consolidate the products based on product_category_name
- **Order by ()** – To arrange values in descending order

SQL Query -

```
Select Product_Category_Name, Min(Price), Max(Price), Round(Avg(Price))
```

```
from Amazon_Brazil.Order_items I
```

```
Join Amazon_Brazil.Products P
```

```
on I.product_id = P.Product_id
```

```
Group by product_Category_Name
```

```
order by Avg(price) Desc
```

Results –

	product_category_name character varying	min double precision	max double precision	round double precision
1	pcs	34.5	6729	1098
2	portateis_casa_forno_e_cafe	10.19	2899	624
3	eletrodomesticos_2	13.9	2350	476
4	agro_industria_e_comercio	12.99	2990	342
5	instrumentos_musicais	4.9	4399.87	282
6	eletroportateis	6.5	4799	281
7	portateis_cozinha_e_preparadores_de_aliment...	17.42	1099	265
8	telefonia_fixa	6	1790	226
9	construcao_ferramentas_seguranca	8.9	3099.9	209
10	relogios_presentes	8.99	3999.9	201
11	climatizacao	10.9	1599	185

Recommendations

1. Focus on high and mid value categories:

- Categories such as pcs, portateis casa forno e cafe and eletrodomesticos 2 have the highest average prices. Amazon should focus on premium positioning and exclusive deals to attract high-spending customers.
- Categories like agro industria e comercio, instrumentos musicais and portateis cozinha e preparadores de alimentos show average prices in the mid-range. Amazon could explore bundling and cross-selling opportunities to increase sales.

2. Promoting low value categories:

- Categories such as artes, brinquedos, and papelaria have relatively lower average prices. Focusing on volume sales through discounts or bulk purchase offers would be effective for these segments.

Analysis 2.3

Problem Statement

Amazon India wants to identify the customers who have placed multiple orders over time. Find all customers with more than one order, and display their customer unique IDs along with the total number of orders they have placed.

Table Used : Amazon_Brazil.Customers, Amazon_Brazil.Orders

Columns Used : Unique_id, Order_id

Functions Used :

- Join()** – To combine Amazon_Brazil.Customers and Amazon_Brazil.Orders table
- Count ()**– To count the number of orders placed

SQL Query -

```
Select customer_Unique_id, count(order_id) as Total_orders  
from Amazon_Brazil.Customers C  
Join Amazon_Brazil.Orders O  
ON C.Customer_id = O.Customer_id  
Group by customer_unique_id  
Having count(order_id) > 1
```

Results –

Sql query returned 3410 rows. Here 10 rows are displayed.

	customer_unique_id character varying (200)	total_orders bigint
1	00172711b30d52eea8b313a7f2cce...	2
2	004288347e5e88a27ded2bb237470...	2
3	004b45ec5c64187465168251cd1c9...	2
4	0058f300f57d7b93c477a131a59b36...	2
5	00a39521eb40f7012db50455bf0834...	2
6	00cc12a6d8b578b8ebd21ea4e2ae8...	2
7	011575986092c30523ecb71ff10cb4...	2
8	011b4adcd54683b480c4d841250a9...	2
9	012452d40dafae4df401bcd74cdb4...	2
10	012a218df8995d3ec3bb221828360...	2

Recommendations

There are total 3140 customers who have placed multiple orders from 2 to 16 each. Offering cashbacks and free gifts can help to retain them and encourage them to shop more.

Analysis 2.4

Problem Statement

Amazon India wants to categorize customers into different types ('New – order qty. = 1'; 'Returning' –order qty. 2 to 4; 'Loyal' – order qty. >4) based on their purchase history. Use a temporary table to define these categories and join it with the customers table to update and display the customer types.

Table Used : Amazon_Brazil.Customers, Amazon_Brazil.Orders

Columns Used : Unique_id, Order_id

Functions Used :

- **Join()** – To combine Amazon_Brazil.Customers and Amazon_Brazil.Orders table
- **Count ()**– To count the number of orders placed

SQL Query -

```
create temporary table customer_categories as
select customer_id,
case
when count(order_id) = 1 then 'New'
when count(order_id) between 2 and 4 then 'Returning'
when count(order_id) > 4 then 'Loyal'
else 'NA'
end as customer_type
from amazon_brazil.orders
group by customer_id;
select c.customer_id, cc.customer_type
from amazon_brazil.customers c
join customer_categories cc
on c.customer_id = cc.customer_id;
```

Results -

Here 10 rows are displayed.

	customer_id character varying (200)	customer_type text
1	00012a2ce6f8dcda20d059ce98491...	New
2	000161a058600d5901f007fab4c271...	New
3	0001fd6190edaaf884bcdf3d49edf079	New
4	0002414f95344307404f0ace7a26f1...	New
5	000379cdec625522490c315e70c7a...	New
6	0004164d20a9e969af783496f34086...	New
7	000419c5494106c306a97b5635748...	New
8	00046a560d407e99b969756e0b10f...	New
9	00050bf6e01e69d5c0fd612f1bcfb69c	New
10	000598caf2ef4117407665ac332751...	New
11	00062b33cb9f6fe976afdcff967ea74d	New
12	00066ccbe787a588c52bd5ff404590...	New
13	00072d033fe2e59061ae5c3aff1a2b...	New
14	0009a69b72033b2d0ec8c69fc70ef7...	New
15	000bf8121c3412d3057d32371c5d3...	New

Recommendations

1. Retaining Loyal and Returning customers:
 - There are 98 Loyal (more than 4 orders) and 117 Returning (2 - 4 orders). To retain them Amazon could offer cashback point or free delivery on every purchase.
2. Focus on New customers:
 - Maximum number (98133) of customers are 'New' (1 order). To encourage them to shop more amazon could provide best deals and offer

Analysis 2.5

Problem Statement

Amazon India wants to know which product categories generate the most revenue. Use joins between the tables to calculate the total revenue for each product category. Display the top 5 categories.

Table Used : Amazon_Brazil.Products, Amazon_Brazil.Orders_items

Columns Used : Product_category_name, Price

Functions Used :

- **Join()** – To combine Amazon_Brazil.Products and Amazon_Brazil.Order_items
- **Group by ()** – To consolidate the total revenue based on product_category_name
- **Limit ()** – To find the top “n” values

SQL Query -

```
select p.product_category_name, round(sum(o.price)) as total_revenue  
from amazon_brazil.products p  
join amazon_brazil.order_items o  
on p.product_id = o.product_id  
group by product_category_name  
order by total_revenue desc  
limit 5;
```

Results -

	product_category_name character varying	total_revenue double precision
1	beleza_saude	1257865
2	relogios_presentes	1203060
3	cama_mesa_banho	1032269
4	esporte_lazer	985881
5	informatica_acessorios	910605

Recommendations

- Categories like Beleza saude, relogios presentes, and cama mesa banho etc should be focused, as they generate the highest revenue. Consider targeted campaigns and loyalty programs to retain and expand customer bases in these areas.
- Ensure a strong supply chain for these high-revenue categories. Maintaining adequate stock levels and reducing delivery times could help increase customer satisfaction and sales.

Analysis 3.1

Problem Statement

The marketing team wants to compare the total sales between different seasons. Use a subquery to calculate total sales for each season (Spring, Summer, Autumn, Winter) based on order purchase dates, and display the results. Spring is in the

months of March, April and May. Summer is from June to August and Autumn is between September and November and rest months are Winter.

Table Used : Amazon_Brazil.Orders, Amazon_Brazil.Orders_items

Columns Used : Order_Purchase_Timestamp, Price

Functions Used :

- **Case()** – To segment months as particular season.
- **Join ()** – To combine table Amazon_Brazil.Orders, Amazon_Brazil.Orders_items

SQL Query -

```
select season, round(sum(oi.price)) as total_sales
from
(
  select o.order_id,
  case
    when extract(month from order_purchase_timestamp) in (3, 4, 5) then 'Spring'
    when extract(month from order_purchase_timestamp) in (6, 7, 8) then 'Summer'
    when extract(month from order_purchase_timestamp) in (9, 10, 11) then 'Autumn'
    else 'Winter'
  end as season
  from amazon_brazil.orders o
) as order_season
join amazon_brazil.order_items oi
on order_season.order_id = oi.order_id
group by season;
```

Results -

	season text	total_sales double precision
1	Winter	2905750
2	Autumn	2348813
3	Spring	4216722
4	Summer	4120360

Recommendations

- Spring and summer generate the highest sales. Prioritize major promotions, new product launches, and discounts during these seasons to increase consumer spending.
- Autumn shows comparatively lower sales. Targeted promotions, seasonal deals and festival sales can help to boost demand in this season.
- Winter sales are relatively strong. Introducing end-of-year sales, holiday promotions, and winter-specific product lines can further drive revenue during this season.

Analysis 3.2

Problem Statement

The inventory team is interested in identifying products that have sales volumes above the overall average. Write a query that uses a subquery to filter products with a total quantity sold above the average quantity.

Table Used : Amazon_Brazil.Orders_items

Columns Used : Product_id, Order_id

Functions Used :

- **Count ()** – To count the quantity sold.
- **Group by** – To consolidate values based on a parameter.

SQL Query –

```
select product_id, total_quantity_sold
from
(select product_id, count(distinct order_id) as total_quantity_sold
from amazon_brazil.order_items
group by product_id
) as product_totals
where total_quantity_sold >
(select avg(total_quantity_sold)
from
(select product_id, count(distinct order_id) as total_quantity_sold
from amazon_brazil.order_items
```

```

group by product_id

) as avg_totals

)

order by total_quantity_sold desc

```

Results –

SQL Query returned 5538 rows, 15 rows are listed here

	product_id character varying (200)	total_quantity_sold bigint
1	99a4788cb24856965c36a24e339b6...	467
2	aca2eb7d00ea1a7b8ebd4e6831466...	431
3	422879e10f46682990de24d770e7f8...	352
4	d1c427060a0f73f6b889a5c7c61f2ac4	323
5	389d119b48cf3043d311335e499d9c...	311
6	53b36df67ebb7c41585e8d54d6772e...	306
7	368c6c730842d78016ad823897a37...	291
8	53759a2ecddad2bb87a079a1f1519f...	287
9	154e7e31ebfa092203795c972e5804...	269
10	2b4609f8948be18874494203496bc...	259
11	3dd2a17168ec895c781a9191c1e95...	255
12	7c1bd920dbdf22470b68bde975dd3c...	225
13	e0d64dcfaa3b6db5c54ca298ae101d...	194
14	5a848e4ab52fd5445cdc07aab1c40e...	194
15	bb50f2e236e5eea010068013765468...	187

Recommendations

- Products with higher sales volumes should be prioritized for inventory restocking. This will avoid stockouts and meet ongoing customer demand.
- Analyse historical trends for high-demand products to forecast future demand. This can help optimize procurement and logistics.
- Monitor low - demand products - Reducing stock or offer discounts to clear slower - moving inventory.

Analysis 3.3

Problem Statement

To understand seasonal sales patterns, the finance team is analysing the monthly revenue trends over the past year (year 2018). Run a query to calculate total revenue generated each month and identify periods of peak and low sales. Export

the data to Excel and create a graph to visually represent revenue changes across the months.

Table Used : Amazon_Brazil.Orders_items, Amazon_Brazil.Orders

Columns Used : Price, Order_Purchase_Timestamp

Functions Used :

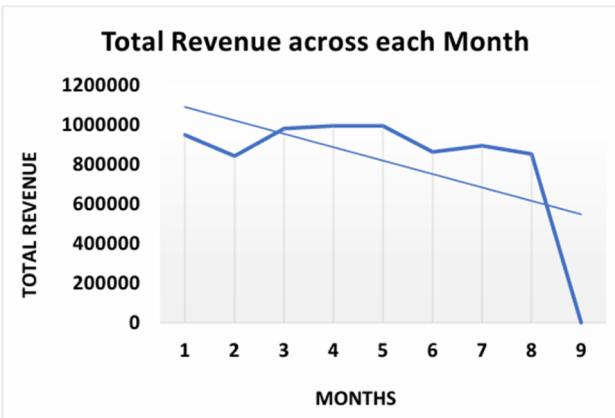
- **Extract ()** – To extract month from date.
- **Join()** – To combine Amazon_Brazil.Orders_items and Amazon_Brazil.Orders table
- **Group by** – To consolidate revenue based on month

SQL Query –

```
select extract(month from order_purchase_timestamp) as month,  
round(sum(oi.price)) as total_revenue  
from amazon_brazil.orders o  
join amazon_brazil.order_items oi  
on o.order_id = oi.order_id  
where extract(year from (order_purchase_timestamp)) = 2018  
group by month;
```

Results –

	month numeric	total_revenue double precision
1	1	950030
2	2	844179
3	3	983213
4	4	996648
5	5	996518
6	6	865124
7	7	895507
8	8	854686
9	9	145



Recommendations

- March, April, and May show the highest revenue, each exceeding 950,000 BRL. These months should be targeted for major promotional campaigns to capitalize on high demand.
- February and August exhibit comparatively low sales. Introduce strategic promotions or limited-time offers during these periods to stimulate demand.

Analysis 3.4

Problem Statement

A loyalty program is being designed for Amazon India. Create a segmentation based on purchase frequency: 'Occasional' for customers with 1-2 orders, 'Regular' for 3-5 orders, and 'Loyal' for more than 5 orders. Use a CTE to classify customers and their count and generate a chart in Excel to show the proportion of each segment.

Table Used : Amazon_Brazil.Orders

Columns Used : Customer_id, Order_id

Functions Used :

- **Case ()** – To segment Customer type
- **Group by** – To group by customer type

SQL Query –

with customer_segmentation as

(

```
select customer_id, count(order_id) as count,
case
when count(order_id) > 5 then 'Loyal'
when count(order_id) between 3 and 5 then 'Regular'
when count(order_id) between 1 and 2 then 'Ocassional'
else 'NA'
end as customer_type
from amazon_brazil.orders
group by customer_id
)
select customer_type, count(customer_type) as count from customer_segmentation
group by customer_type;
```

Results -

	customer_type	count
1	Ocassional	98144
2	Regular	106
3	Loyal	98



Recommendations

- Majority of customers (98,144) fall into the Occasional segment. Target this group with personalized offers, discounts, and incentives to encourage repeat purchases and converting them into Regular segment.
- Regular segment (106 customers) can be further nurtured into becoming Loyal. Rewards can encourage them to increase their purchase frequency, such as discounts after a certain number of orders or additional benefits for higher spending.
- Loyal customer segment is relatively small (98 customers), they represent the most engaged users. Design exclusive loyalty rewards, early access to sales, and premium services to retain their loyalty and drive long-term value.

Analysis 3.5

Problem Statement

Amazon wants to identify high-value customers to target for an exclusive rewards program. You are required to rank customers based on their average order value (avg_order_value) to find the top 20 customers.

Table Used : Amazon_Brazil.Orders, Amazon_Brazil_Order_items

Columns Used : Customer_id, Order_id

Functions Used :

- Avg ()** – To get average order value
- Rank ()** – To rank customers based on order value in descending order
- Limit ()** – To get top “n” values

SQL Query –

```
select o.customer_id, avg(oi.price) as avg_order_value,  
rank () over (order by avg(oi.price) desc) as customer_rank  
from amazon_brazil.orders o  
join amazon_brazil.order_items oi
```

```

on o.order_id = oi.order_id
group by o.customer_id
order by avg_order_value desc
limit 20;

```

Results -

	customer_id character varying (200)	avg_order_value double precision	customer_rank bigint
1	c6e2731c5b391845f6800c97401a...	6735	1
2	f48d464a0baaea338cb25f816991ab...	6729	2
3	3fd6777bbce08a352fddd04e4a7cc8...	6499	3
4	df55c14d1476a9a3467f131269c24...	4799	4
5	24bbf5fd2f2e1b359ee7de94defc4a15	4690	5
6	3d979689f636322c62418b6346b1c...	4590	6
7	1afc82cd60e303ef09b4ef9837c950...	4399.87	7
8	35a413c7ca3c69756cb75867d6311...	4099.99	8
9	e9b0d0eb3015ef1c9ce6cf5b9dcbee...	4059	9
10	c6695e3b1e48680db36b487419fb0...	3999.9	10
11	926b6a6fb8b6081e00b335edaf578...	3999	11
12	3be2c536886b2ea4668eced3a80dd...	3980	12
13	31e83c01fce824d0ff786fc48dad009	3930	13
14	eb7a157e8da9c488cd4ddc48711f1...	3899	14
15	19b32919fa1198aefc0773ee2e46e6...	3700	15

Recommendations

- Exclusive reward program should be there for the identified top 20 customers. Offer tailored benefits such as premium customer service, early access to sales, or personalized product recommendations.
- Loyalty reinforcement can be done by providing personalized offers, loyalty points or limited-edition product launches to keep them connected to the brand.
- Increase retention through incentives like discounts or cashback on future purchases.

Analysis 3.6

Problem Statement

Amazon wants to analyze sales growth trends for its key products over their lifecycle. Calculate monthly cumulative sales for each product from the date of its first sale. Use a recursive CTE to compute the cumulative sales (total_sales) for each product month by month.

Table Used : Amazon_Brazil.Orders, Amazon_Brazil_Order_items

Columns Used : Customer_id, Order_id

Functions Used :

- **Extract ()** – To get month from date
- **Sum ()** – To get the sum of price
- **Join ()** – To join Amazon_Brazil.Orders and Amazon_Brazil_Order_items

SQL Query –

```
with monthly_sales as
(
  select oi.product_id,
         extract(month from o.order_purchase_timestamp) as sale_month,
         sum(oi.price) as monthly_sale
    from amazon_brazil.orders o
   join amazon_brazil.order_items oi
     on o.order_id = oi.order_id
   group by product_id, sale_month
)
```

Results –

Sql Query returned 60796 rows. Here 15 rows are shown

	product_id character varying (200)	sale_month numeric	total_sales double precision
1	00066f42aeeb9f3007548bb9d3f33c38	5	102
2	00088930e925c41fd95ebfe695fd2655	12	130
3	0009406fd7479715e4bef61dd91f2462	12	229
4	000b8f95fc9e0096488278317764d...	8	118
5	000d9be29b5207b54e86aa1b1ac54...	4	199
6	0011c512eb256aa0dbbb544d8dffcf6e	12	52
7	00126f27c813603687e6ce486d909d...	9	498
8	001795ec6f1b187d37335e1c470476...	10	39
9	001795ec6f1b187d37335e1c470476...	11	117
10	001795ec6f1b187d37335e1c470476...	12	350
11	001b237c0e9bb435f2e54071129237...	8	79
12	001b72dfd63e9833e8c02742adf472...	2	105
13	001b72dfd63e9833e8c02742adf472...	3	140
14	001b72dfd63e9833e8c02742adf472...	7	245
15	001b72dfd63e9833e8c02742adf472...	8	315

Recommendations

- Through promotions and discounts retain the high sales of these products.
- Offer large discounts on bulk purchase to increase sale volume and revenue.
- Offer free shipping, clearance sale discounts, clubbing with high sale products etc can help enhance the sales of these products.