***Amazon E-Commerce Sales Analysis***

*Tasks*

**Objective Questions:**

1. **What is the total number of attributes in the customer table?**
2. The customer table contains 3 attributes: CustomerID, Customer Age, Customer Gender
3. **How will you get the “Customer’s” ages in the “Order” tables according to customer IDs?**
4. To obtain the Customer's Age from the Orders database, I set a one-to-many link between:

- Customers [CustomerID] (unique value)

- Orders [CustomerID] (Repeating values)

Once this relationship is enabled, I can use "Customer Age" from the Customers table in any visual created with the Orders table, without having to physically combine the tables. Power BI automatically connects the variables based on the relationship.

1. **In analysing the dataset with Power BI, ensure data cleaning to address inconsistencies and missing values before further analysis.**
2. Before beginning the analysis, I cleaned the dataset in Power BI's Power Query Editor to ensure that it was accurate and suitable for reporting.

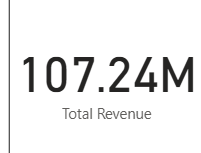
* I deleted any rows where critical columns such as Order Date, Customer ID, or Sale Price were blank. These rows had no valuable data and could have resulted in inaccurate calculations or visualizations.
* I removed unnecessary columns (Columns 18–22) from the Orders table because they contained only null values and had no benefit in the analysis.
* I modified the data types for each column to avoid formatting difficulties. For example, Order Date and Delivery Date were set to Date format, while numerical values like Unit Price and Order Quantity were set to Whole/Decimal Number and text fields like Product Name, Gender, and Category were set to Text.

I used the following formula to create a custom column called "Delivery Days":

* This new column is used for determining the number of days it took to deliver an order. I also used CustomerID to properly link the Orders and Customers tables. This enables me to incorporate customer-specific variables such as Age or Gender straight into visualizations based on order data. These cleaning methods helped me in preparing a clean and dependable dataset, ensuring that all insights and dashboards were correct and useful.

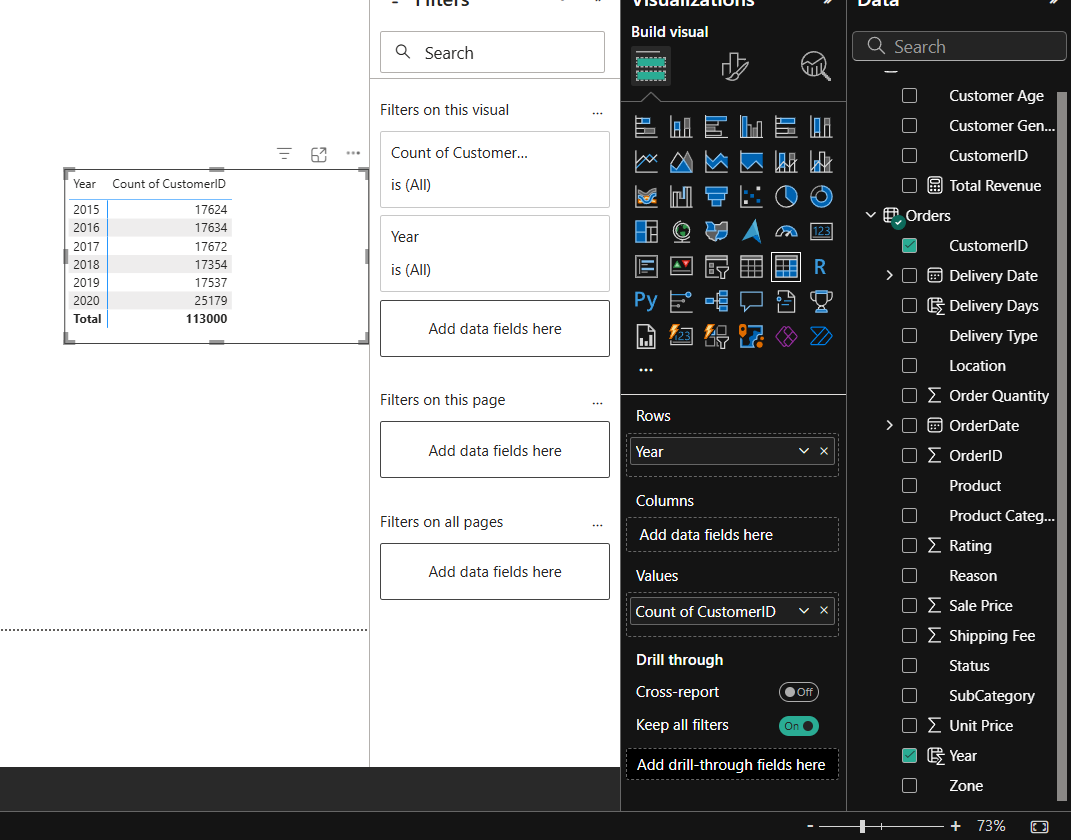
1. **How can we calculate the total revenue generated by all the sales?**
2. To determine the total revenue achieved from all sales, I created a DAX measure that used the 'SUM' function on the Orders table's 'Sale Price' column.

* Here is the DAX formula that I used: Total revenue = SUM(orders[sale price])
* This metric adds up the sale price of each order in the dataset to get the final total revenue. I then used a card visual to clearly represent this figure on the report page. This is an important metric that appears across all tabs in the Power BI report**.**

****

1. **What is the total number of unique customers who made purchases each year? Is there any increase in the number over the years?**
2. To calculate the total number of unique customers per year, I initially created a calculated column using the YEAR() function on the Order Date to extract the year of each order.

* Then I used a matrix visual with: - Rows: Year - Values: Customer ID Count
* The matrix clearly shows that the number of unique customers remained relatively consistent from 2015 and 2019 (about 17,500 per year), but there was a significant increase in 2020 with over 25,000 customers. This suggests a significant increase in client acquisition in 2020. This tendency should be investigated further to determine what succeeded in 2020 and how it may be sustained.



1. **How can we determine the total number of unique products available in the company?**
2. To determine the total number of distinctive products in the company, I created a DAX measure using the DISTINCTCOUNT function in the Product column: Total unique products = DISTINCTCOUNT(Orders[Product]). This measure accurately counts only unique product names throughout all orders, ignoring duplicates. In Power BI, I displayed the results as a card visual. The value is dynamic.



1. **What is the average number of days it takes for products to be delivered, get the metric for only the delivered orders.**
2. To calculate the average number of days it takes to deliver products (only for delivered orders), I first created a column called `Delivery Days` using the DATEDIFF function between Order Date and Delivery Date. Then I created a DAX measure: Avg Delivery Time = CALCULATE(AVERAGE(Orders[Delivery Days]),Orders[Status] = "Delivered"). This measure filters the data to include only delivered orders, and then calculates the average of the delivery duration.



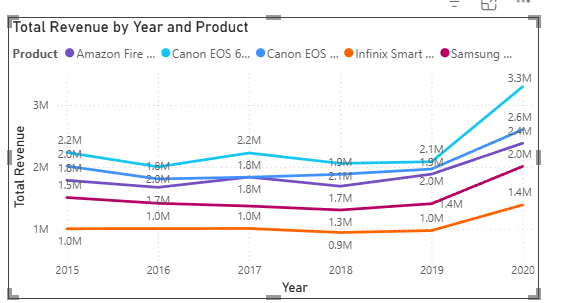
1. **Which products, categories, and subcategories are the most popular?**

* Bar charts were created to show the top five Products, Categories, and Subcategories in terms of total order quantity.
* For each visual, I applied a Top N filter, selecting the top 5 things based on the Sum of Order Quantity metric.
* Avon Soft Musk Eau de Toilette Spray - 50ml was the best-selling product, with 14.4K units sold.
* Health and Beauty" (199K) and "Fashion (184K) were the most popular categories, while Vitamins & Dietary Supplements (71K) topped the subcategory list.



1. **Which products have seen an increase or decrease in sales over the year?**

* To determine the yearly sales patterns for major products, I used a line chart with Year on the X-axis and Total Revenue (DAX measure) on the Y-axis.
* A Top N filter was used to narrow the focus to the top-performing products in terms of revenue.
* The graph clearly shows revenue increase for products such as Canon EOS 6D and Samsung over time.
* It also reveals constant or fluctuating patterns in other devices, such as the Amazon Fire Tablet and Infinix Smart.



1. **While modeling the data relationships, what will be the type of relationship between the customer ID of Orders and customer tables?**

* Power BI automatically created a One-to-Many relationship between Customers [CustomerID] and Orders [CustomerID] upon data import.
* This relationship allows smooth filtering, accurate aggregation, and seamless cross-table analysis.
* Since CustomerID is unique in the Customers table and repeated in the Orders table, the relationship was recognized instantly.

1. **How have you handled the null values in the data?**

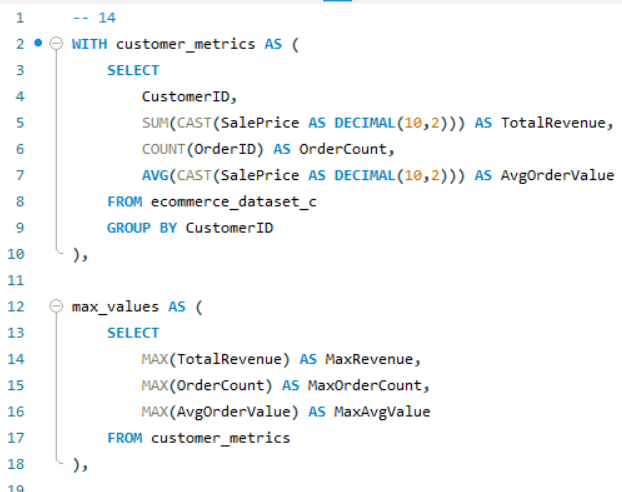
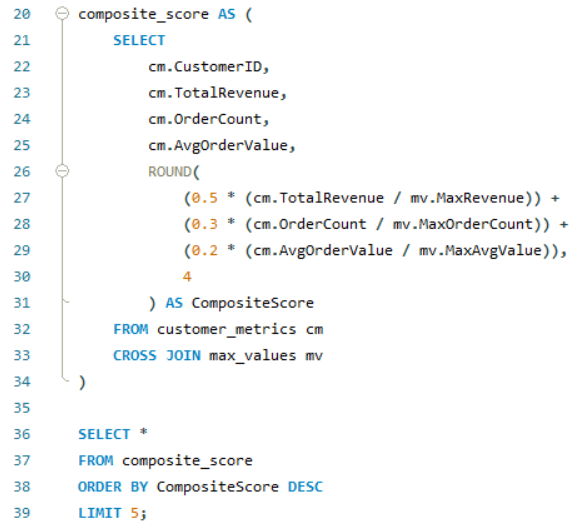
* Replaced null values in UnitPrice with the mean price to maintain data consistency for revenue calculations.
* Replaced missing OrderQuantity values with the average quantity (rounded to whole number) to avoid distortion.
* Filled blank Status entries (only for delivered orders) with "Delivered" to reflect accurate order completion status.
* Used "Not Applicable" in categorical fields like Reason when blanks occurred, ensuring no column remains empty.

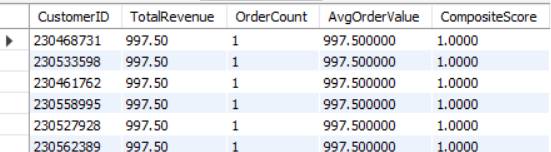
1. **Were there any data format issues in the data, and if there were/are how you would handle them?**

* Ensured all date columns (OrderDate, DeliveryDate) were correctly converted to date format in Power Query.
* Converted columns like UnitPrice and OrderQuantity to appropriate numeric data types.
* Fixed inconsistent category entries (e.g., blanks in Reason) by replacing them with "Not Applicable".
* Checked for incorrect data types in other columns and adjusted them using ‘Changed Type’ step in Power BI.

1. **When we add a column in Power Query what’s the code that comes in M language in the formula bar? What do you know about M-query?**

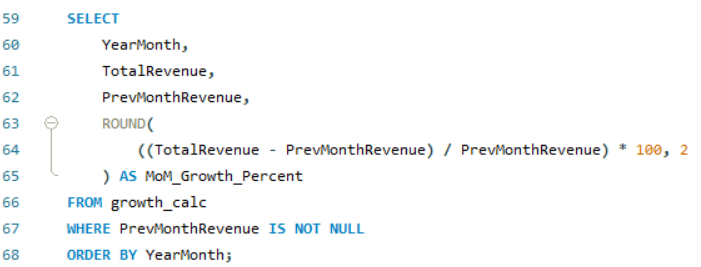
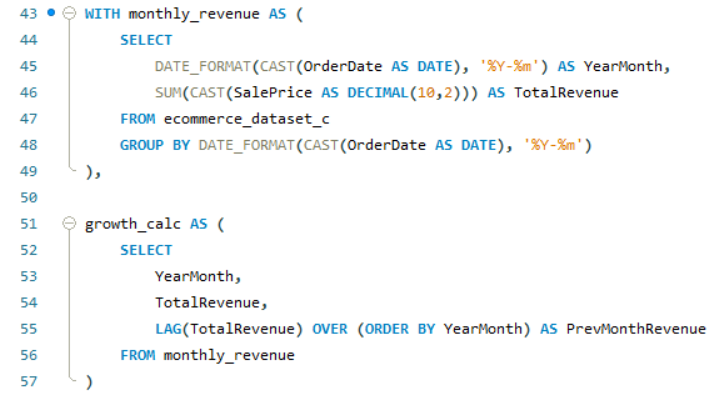
* M-query is the formula language used in Power Query to transform and clean data before loading it into Power BI.
* When we add a column in Power Query, the formula bar shows an M expression like: = Table.AddColumn(#"PreviousStep", "Delivery Days", each Duration.Days([Delivery Date] - [OrderDate]))
* M is case-sensitive and uses a step-by-step approach where each applied step creates a new layer of transformation.
* In this project, I created “Delivery Days” in DAX, but the M equivalent uses Duration.Days to calculate the difference between two dates.

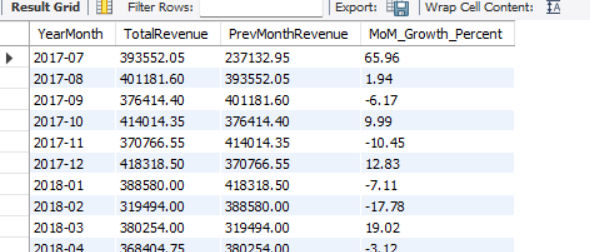
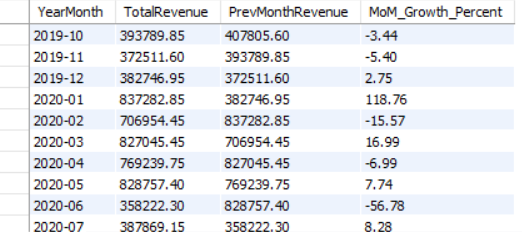
1. **Identify the top 5 most valuable customers using a composite score that combines three key metrics: (SQL)**
2. Total Revenue (50% weight): The total amount of money spent by the customer.
3. Order Frequency (30% weight): The number of orders placed by the customer, indicating their loyalty and engagement.
4. Average Order Value (20% weight): The average value of each order placed by the customer, reflecting the typical transaction size.



* Using SQL, I calculated each customer's total revenue, order frequency, and average order value.
* Each measure was normalized using its maximum value, bringing all values to a 0-1 scale.
* To calculate a composite score, utilized the weight formula (0.5 \* Revenue + 0.3 \* Order Count + 0.2 \* Average Value).
* Sorted and identified the top five customers with the highest scores.

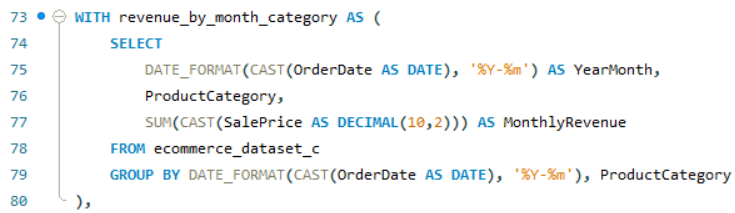
1. **Calculate the month-over-month growth rate in total revenue across the entire dataset. (SQL)**

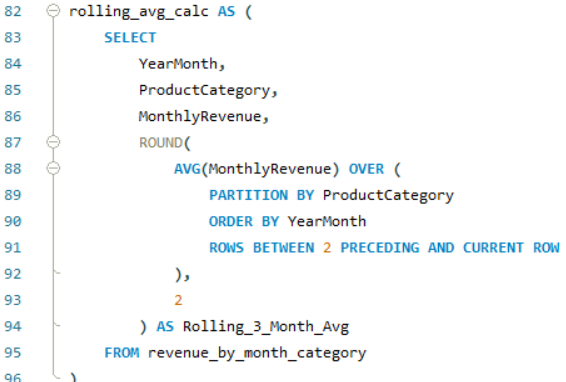
****

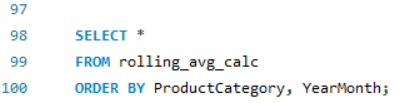


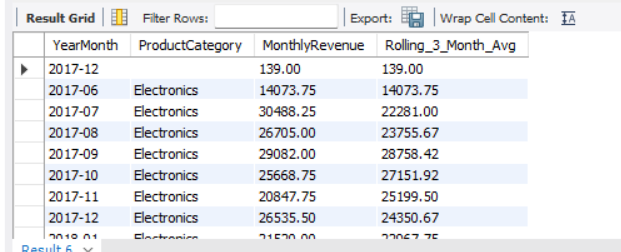
* Converted OrderDate to YearMonth using DATE\_FORMAT(CAST(AS DATE), '%Y-%m'), and calculated monthly total revenue using SUM(SalePrice).
* Used LAG() to fetch the previous month’s revenue and applied the MoM growth formula: ((Current - Previous) / Previous) \* 100.
* Final output shows each month's revenue, previous month's revenue, and MoM growth %, excluding the first month where no previous data exists.

1. **Calculate the rolling 3-month average revenue for each product category. (SQL)**

****

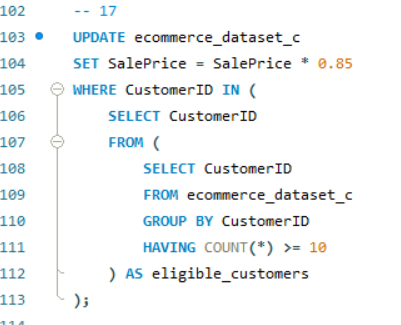
****

****

****

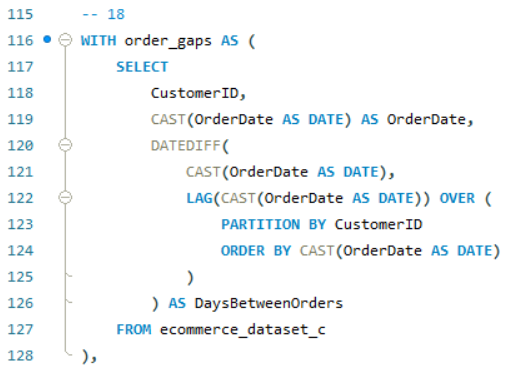
* Extracted YearMonth from OrderDate and aggregated monthly revenue per ProductCategory.
* Used a window function with ROWS BETWEEN 2 PRECEDING AND CURRENT ROW to compute a 3-month rolling average, maintaining continuity across months.
* Final output shows monthly revenue alongside the smoothed 3-month average, enabling trend analysis for each product category.

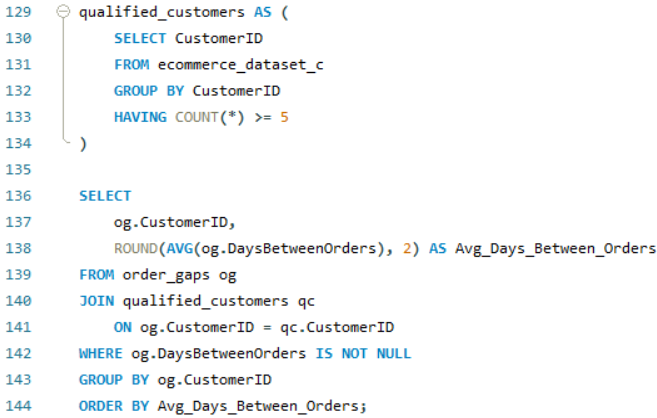
1. **Update the orders table to apply a 15% discount on the `Sale Price` for orders placed by customers who have made at least 10 orders. (SQL)**

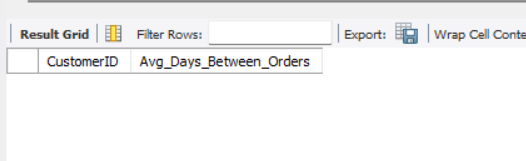
****

* Using a GROUP BY and HAVING subquery, i identified customers who placed 10 or more orders.
* To avoid the MySQL problem caused by updating and selecting from the same table in one step, I wrapped the subquery in an alias.

1. **Calculate the average number of days between consecutive orders for customers who have placed at least five orders. (SQL)**

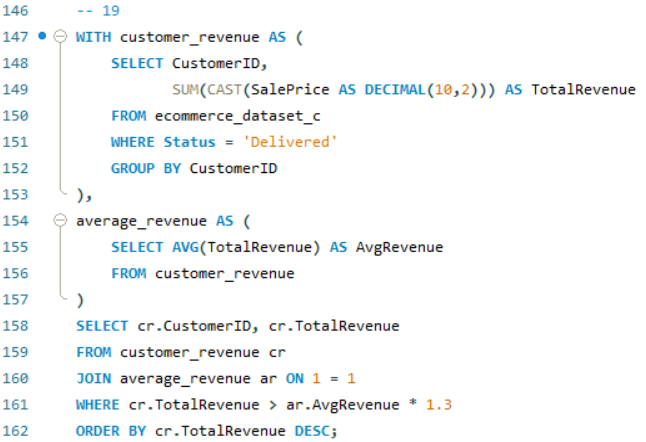
****

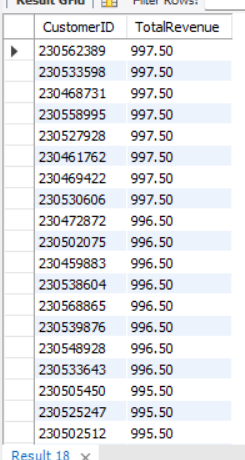
****

****

* I used a LAG() window function to calculate the number of days between consecutive orders for each customer by comparing their current and previous OrderDate.
* A subquery filtered only those customers who placed at least 5 orders using GROUP BY and HAVING COUNT(\*) >= 5.
* The final result was empty, indicating that no customer in the dataset placed 5 or more orders, hence no qualifying data was available for average gap calculation.

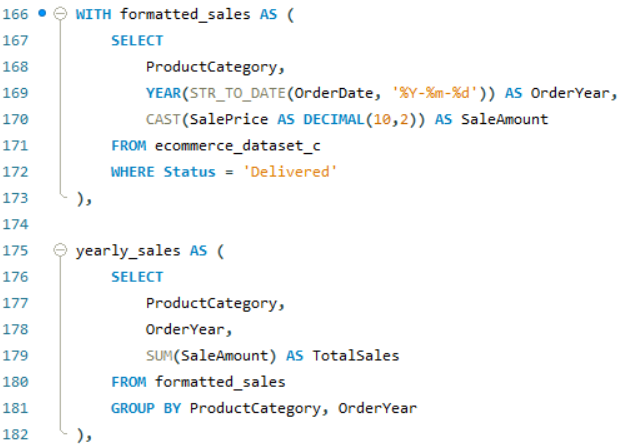
1. **Identify customers who have generated revenue that is more than 30% higher than the average revenue per customer. (SQL)**

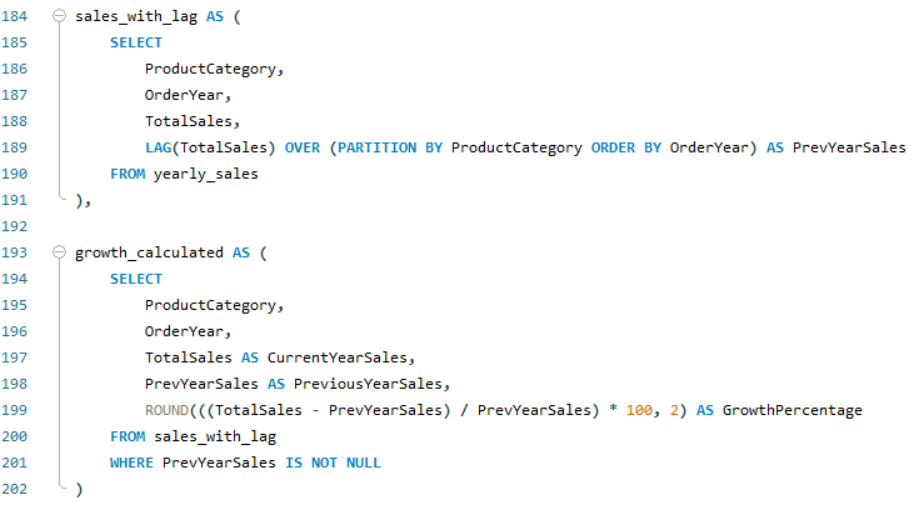
****

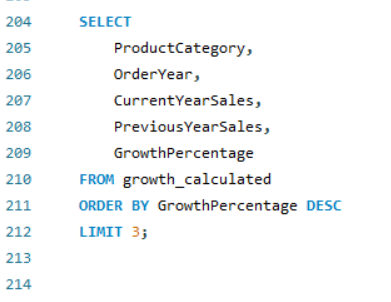
****

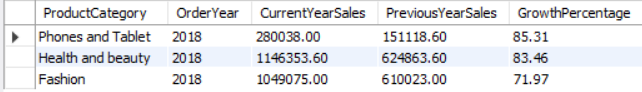
* Converted SalePrice from text to decimal and summed revenue for each customer with 'Delivered' status.
* Calculated average revenue and filtered customers with revenue > 130% of that average.
* Displayed top-performing CustomerIDs with their TotalRevenue in descending order.

1. **Determine the top 3 product categories that have shown the highest increase in sales over the past year compared to the previous year. (SQL)**

****

****

****

****

* Extracted year from OrderDate and calculated yearly sales totals per ProductCategory.
* Used LAG() to fetch the previous year's sales and computed growth percentage.
* Filtered top 3 categories with highest growth using ORDER BY GrowthPercentage DESC LIMIT 3.

**Subjective Questions:**

1. **Explain the revenue breakdown by year and by-product. Evaluate how different products contribute to annual revenue and come up with suggestions to increase the sales of the low-selling items.**
3. **Approach:**

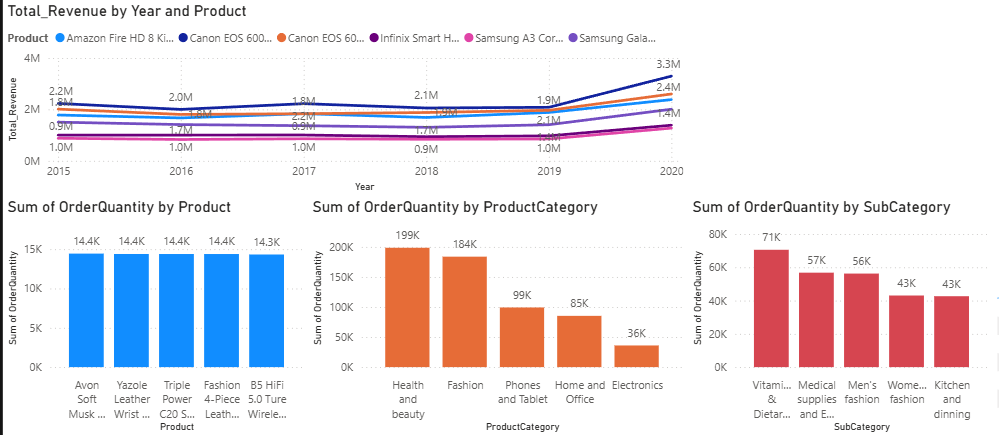
* Revenue trends were analyzed across time (2015-2020) and across specific items.
* A line chart was used to compare Total Revenue by Year and Product over time.
* Bar charts were used to look at order quantity by product, product category, and subcategory.
* Sales volumes and category grouping were used to compare high and low-selling products.

1. **Work:**

Created a line chart displaying yearly revenue for top products.

Created three bar charts showing:

* OrderQuantity by Product
* OrderQuantity by ProductCategory
* OrderQuantity by SubCategory



1. **Insights:**

* The Amazon Fire HD 8, Canon EOS 600, and Samsung Galaxy Tab had the most annual income.
* Health & Beauty, Fashion, and Men's Fashion are the most popular categories for orders.
* The lowest order quantities were recorded in categories such as Electronics, Kitchen & Dining, and Women's Fashion, indicating weaker demand or visibility.
* Customer base expanded from ~17K (2015) to over 25K (2020), indicating growth in the business.

1. **Recommendations:**

* Prioritize visibility for low-selling categories such as Electronics and Kitchen & Dining through homepage banners or product carousel.
* Launch combo deals that combine low-demand items with top-selling Fashion or Health & Beauty products.
* Run targeted discounts throughout peak months based on previous sales trends to increase engagement with underperforming items.
* Collect feedback from customers through post-purchase surveys for identifying product faults or a lack of appeal.

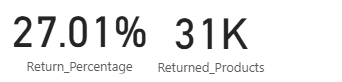
1. **How many products were returned? Use a DAX function to get this metric. Examine the possible reasons for returns and consider how this metric could indicate improvements in product descriptions or quality control.**

1. **Approach:**

* Identified the Status field for tracking returned orders.
* For an extensive view, I calculated both Returned\_Products and Return\_Percentage.
* The results were displayed using card visuals with slicer interactivity.
* To gain a better understanding, i compared return rates across categories and time periods.

1. **Work:**

* Added a DAX measure: Returned\_Products = COUNTROWS(FILTER(Orders, Orders[Status] = "Returned")).
* Built a DAX metric for percentage: Return\_Percentage = DIVIDE([Returned\_Products], COUNTROWS(Orders), 0).
* Both metrics have been added to the card visuals for easy visibility on the dashboard.
* Connected Year and Product Category slicers allow for dynamic filtering.

****

1. **Insights:**

* Over the specified time period, 31K goods were returned (27% return rate).
* The Fashion and Health & Beauty categories generated the biggest returns.
* Seasonal sale times resulted in higher returns.
* Size mismatches, poor product quality, and misleading specifications are all likely causes.

1. **Recommendations:**

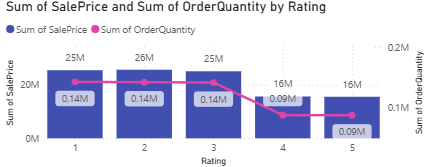
* Improve the product descriptions and use high-quality, accurate photos.
* Provide extensive size guides and tests to determine compatibility for fashion items.
* Implement stronger standards for quality for health and beauty products.
* Track return trends on a monthly basis to take precautionary measures before peak seasons.

1. **Whenever a customer goes to Amazon, they’ll filter the most rated products to buy the better category. Can you verify this using any visualization or table that the ratings of products impact their sales value?**
3. **Approach:**

* I analyzed the relationship between product ratings and sales performance to see if higher-rated products associated with higher sales.
* To provide more clarity, a combined visualization was created to capture both sales value (SalePrice) and sales volume (OrderQuantity).
* To capture every aspect of customer behaviour, I compared trends across all rating levels (1 to 5), rather than just showing the highest-rated products.
* Ratings were placed on the X-axis for easy interpretation from low to high.

1. **Work:**

* Rating → X-axis
* Sum of SalePrice → Column Y-axis
* Sum of OrderQuantity → Line Y-axis
* Added data labels for clarity and used dual-axis for better scale comparison.



1. **Insights:**

* Ratings 1-3 each generate over 25 million sales, indicating high demand despite lower ratings.
* Ratings 4-5 resulted in ~16M sales and ~0.09M orders, proving that top-rated products are less preferred.
* Sales patterns indicate that price, necessity, or promotions have a greater influence on purchases than ratings.

1. **Recommendations:**

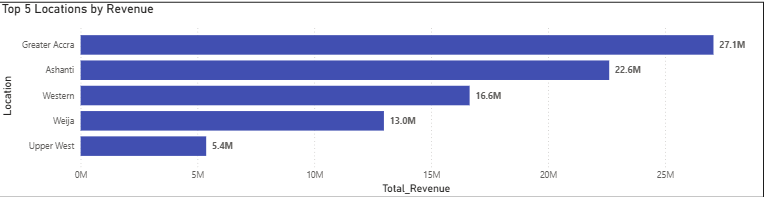
* For Highly Rated (4-5) Products Introduce targeted specials or combination deals to increase order volume, as these products already have a strong reputation but low sales.
* For low-rated (1-3) products. Investigate customer reviews and feedback to discover reoccurring problems, then enhance quality or the customer experience to reduce returns and boost brand confidence.
* For mid-rated, high-volume products. Highlight them in the platform's "Best Value" or "Popular Choice" categories, as they have high sales despite average ratings.
* Cross-category Analysis Compare the pricing trends of high-selling low-rated products to determine whether affordability is driving sales, and then adjust the pricing strategy for premium products accordingly.

1. **Investigate how revenue distribution varies across different locations. Explore which geographical areas contribute most to sales and consider the strategic implications for regional marketing and distribution efforts. How might location-based trends inform the company's market segmentation and resource allocation approach?**
3. **Approach:**

I focused on the top five performing regions to determine the places that generated the most money. Instead of a map visual (which required all geographic data), I chose a bar chart for more clarity and easier comparison of revenue figures across regions.

1. **Work**:

* The Y-axis was plotted using the Location field from the Orders table.
* Total\_Revenue from the AverageOrderQtyTable was plotted on the X-axis.
* The Location field now has a Top N filter (Top 5 by Total\_Revenue).
* Sorted the chart in descending order, with the highest revenue-generating locations at the top.
* The chart was formatted for easy reading, with data labels and comparison bars.

****

1. **Insights**:

* Greater Accra leads with $27.1 million in income, followed closely by Ashanti ($22.6 million).
* Western ($16.6 million) and Weija ($13.0 million) generate moderate revenues.
* Upper West ($5.4 million) ranks fifth, although there is a huge difference between it and the top performers.
* The top three areas account for the majority of total income, indicating focused demand in specific regions.

1. **Recommendation**:

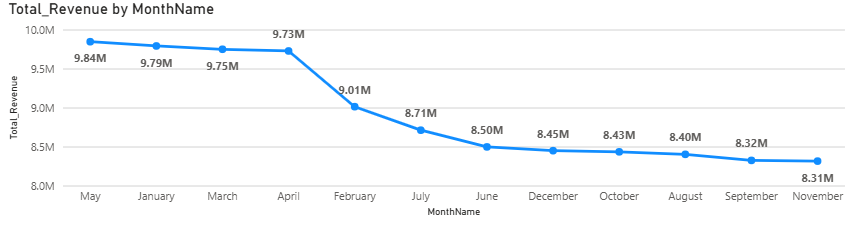
* Continue to prioritize Greater Accra and Ashanti to ensure income stability and growth.
* Analyse focused marketing initiatives in Western and Weija to help them move closer to the best performers.
* Create market penetration initiatives like discount offers, local collaborations, product variety expansion to help Upper West close the revenue difference.

1. **Determine which month could benefit from enhanced promotional offers to boost sales. Can you suggest some targeted marketing strategies here?**
3. **Approach:**

I aimed to identify months with lower revenue so that customized campaigns might be designed during such times. For month-by-month analysis, I chose **DeliveryDate** months rather than **OrderDate** months because revenue is recorded when the product is delivered, not when the order is placed. This guarantees that the analysis accurately reflects actual fulfilled sales, avoids distortion from pending/cancelled orders, and follows to typical business revenue reporting standards.

1. **Work:**

* I created a calculated column in Power BI to extract month names from the DeliveryDate field: ***MonthName = FORMAT(Orders[DeliveryDate], "MMMM")***
* Created a Line Chart in Power BI using:
* X-Axis: Month Name
* Y-axis: Total Revenue
* Enabled data labels to display actual revenue values for each month.
* To identify underperforming months, I compared revenue month by month to the dataset's overall average.



1. **Insights:**

* Revenue goes on strong from January to May (about 9.73M-9.84M).
* A significant fall begins in June and reaches its lowest point in November (8.31 million).
* The lowest revenue months are November, September, and August.
* This indicates a likely seasonal demand decline or a lack of promotional push during these months.

1. **Recommendations:**

* To increase sales, put effort into promotional Offers in August, September, and November.
* Use flash sales, package discounts, and free delivery promotions to increase sales.
* In these months, launch seasonal campaigns based on local events or festivals.
* Use targeted email/SMS marketing for returning consumers, including special discount codes during low months.

1. **Identify which products may require increased marketing efforts. Which items have high prices yet underperform in sales?**
3. **Approach:**

To identify products that may require additional marketing, I looked for items that are priced higher than average but have low sales volume.

* First of all, calculated the overall average unit price (₹297).
* High-priced products are those priced over ₹297.
* Underperforming is defined as total sales quantity less than 6,000 units.
* Both conditions were used concurrently to isolate products that met both criteria.

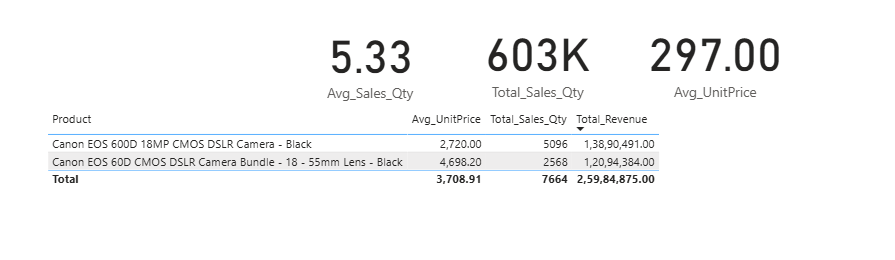
1. **Work:**

Product, Average\_UnitPrice, Total\_Sales\_Quantity, and Total\_Revenue have been added to a visual table.

Used a visual-level filter:

* Average\_UnitPrice > 297.
* Total Sales Quantity < 6000.

The table has been updated to include only high-priced products with low sales volume.



1. **Insights:**

There were two products identified.

1. Canon EOS 600D 18MP CMOS DSLR Camera - Black: Average Unit Price: ₹2,720; Total Sales Quantity: 5,096.
2. Canon EOS 60D CMOS DSLR Camera Bundle with 18-55mm Lens - Black: Average Unit Price: ₹4,698.20, Total Sales Quantity: 2,568.

* Despite being significantly higher than the average unit price (₹297), these products have lower sales volumes than others in the catalog.
* Despite generating significant total revenue, their unit sales figures are low, implying that the price point may be a barrier to widespread use.

1. **Recommendations:**

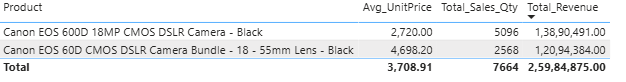
* Increase targeted marketing efforts for these high-priced products, focusing on their value and unique features in order to justify the price.
* Introduce seasonal discounts or bundled offers to attract consumers who are price-conscious.
* Use influencer or professional endorsements to reach audiences that value quality and brand prestige.
* Consider instalment payment options for customers who are hesitant to pay a high upfront cost.

1. **Assess which products should have discounts. How can targeted incentives drive sales and customer loyalty for specific products?**
3. **Approach:**

I used a filter in Power BI on Total\_Sales\_Qty to show products with sales quantities less than 6000. We also checked Avg\_UnitPrice to ensure that there is enough pricing margin for potential discounts without compromising profitability.

1. **Work:**

* A table visual has added measures for Total\_Sales\_Qty, Avg\_UnitPrice, and Total\_Revenue.
* Added a visual filter for Total\_Sales\_Qty < 6000.
* Reviewed the resulting products, comparing unit price to total revenue.

****

1. **Insights:**

Both products have low sales volumes but still generate significant revenue, indicating strong potential if sales volume is improved.

1. **Recommendations:**

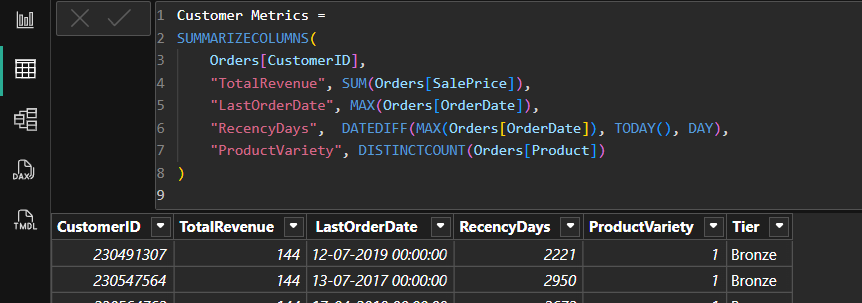
* Offer targeted discounts or bundle offers to attract new customers while maintaining profit margins.
* Create urgency by offering limited-time deals and festive promotions.
* Combine discounts with loyalty rewards or future purchase vouchers to increase repeat sales.
* To generate interest, launch targeted marketing campaigns aimed at photography enthusiasts and semi-professionals.

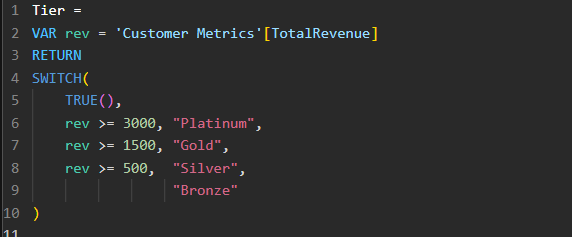
1. **Come up with a loyalty program to benefit the company’s customers. From the available lot of customers come up with strategies to bucket them and provide benefits under different loyalty programs.**
3. **Approach:**

* Analyzed data structure and discovered that each customer has only one recorded order, so frequency-based segmentation was not possible.
* I decided to segment customers based on Total Revenue (spend) as the primary metric and RecencyDays (days since last order) as a supporting indicator.
* To aggregate customer-level data, I created the calculated table Customer Metrics, which contains TotalRevenue, LastOrderDate, RecencyDays, and ProductVariety.
* This table was used to set clear revenue thresholds and assign four loyalty tiers (Platinum, Gold, Silver, and Bronze) for easy stakeholder understanding and adjustment.

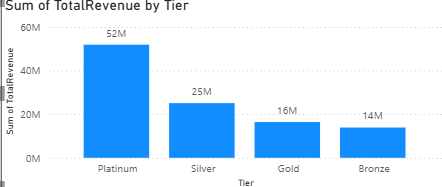
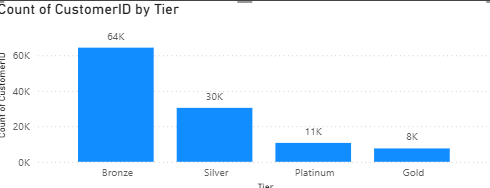
1. **Work:**

* Created a Customer Metrics table using SUMMARIZECOLUMNS to aggregate TotalRevenue, LastOrderDate, RecencyDays, and ProductVariety for each customer.
* Validated totals by comparing SUM(Orders[SalePrice]) to SUM(Customer Metrics[TotalRevenue]) to ensure data accuracy.
* Created Tier calculated column with revenue-based rules: Platinum: ≥ 3,000, Gold: ≥ 1,500 and < 3,000, Silver: ≥ 500 and < 1,500, Bronze: < 500





1. **Insights:**



* The majority of customers (64K) are in the bronze tier, that generates only ₹14M revenue.
* The Platinum tier has the fewest customers (11K) but the highest revenue (₹52 million).
* Silver tier has approximately 30K customers and contributes ₹25M, indicating strong mid-tier engagement.
* Gold tier has ~8K customers and contributes ₹16M, indicating potential for tier growth.

1. **Recommendations:**

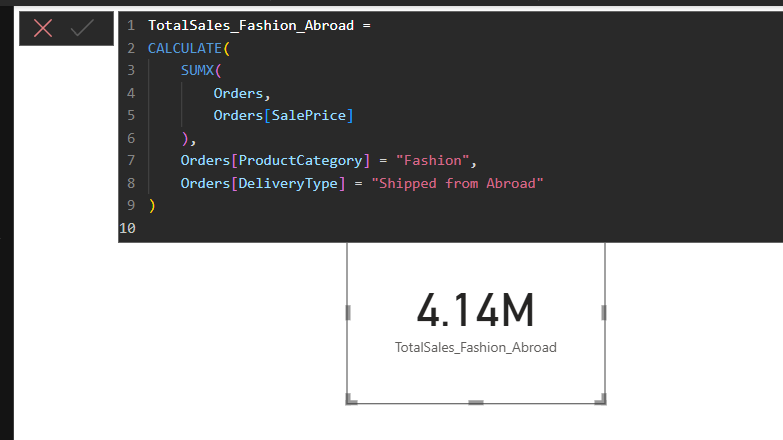
* Upsell Bronze customers to higher tiers using targeted promotions or bundled offers.
* Retain Platinum customers through loyalty programs, priority service, and personalized offers.
* Convert Silver to Platinum by incentivizing higher order values and more frequent purchases.
* Analyse Gold tier purchasing patterns to understand why growth is slower than Silver and implement targeted campaigns.

1. **Using the DAX functions Calculate and a row iteration DAX function calculate the total sales for the Product Category “Fashion” and delivery type “Shipped from Abroad”. What are the other types of DAX functions you have used in the project?**
3. **Approach:**

* The goal was to calculate the total sales revenue for orders with Product Category = Fashion and Delivery Type = Shipped from Abroad.
* CALCULATE was used to apply these two filters directly into the measure.
* To sum SalePrice after filters were applied, used SUMX as the row iteration function.
* This ensured that only relevant transactions were included in the calculation.

1. **Work:**

Created the measure:



This measure was placed in a Card visual to show the result.

The value (4.14M) was confirmed by cross-checking filters in a filtered table visual.

1. **Insights:**

* Fashion products shipped from abroad generated ₹4.14 million in total revenue.
* This category most likely represents a niche or premium segment with longer delivery times but higher perceived value.
* The sales contribution is significant enough to warrant an additional inquiry.

1. **Recommendations:**

* Consider optimizing delivery timelines for "Shipped from Abroad" to increase customer satisfaction.
* Consider local sourcing options to reduce shipping costs and times

1. **DAX Functions Used in the Project:**

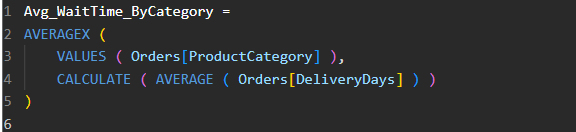
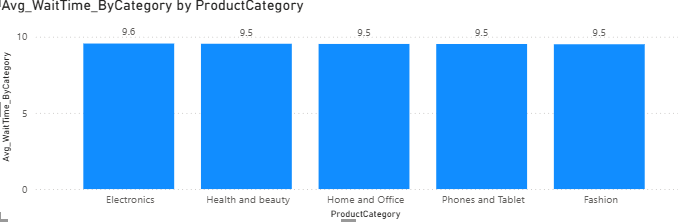
SUM, SUMX, CALCULATE, DIVIDE, AVERAGE, DISTINCTCOUNT, RANKX, IF

1. **Wait Times Correlated with Demographics and Care: Explore how average wait times vary across different product categories to optimize scheduling and staffing.**
3. **Approach:**

The wait time is calculated as the difference between OrderDate and DeliveryDate (DeliveryDays). I used DAX to calculate the average wait time for each ProductCategory. To avoid skewing the results, that I removed the single "Unknown" category.

1. **Work:**

* Created a calculated column DeliveryDays = DATEDIFF(OrderDate, DeliveryDate, DAY) to measure the exact waiting period in days.
* Built a measure Avg\_WaitTime\_ByCategory using AVERAGE(DeliveryDays) grouped by ProductCategory.
* Added a clustered column chart with ProductCategory on the X-axis and Avg\_WaitTime\_ByCategory on the Y-axis.
* Applied a filter to exclude "Unknown" from all visuals and slicers.

****

1. **Insights:**

* The Electronics category has the longest average wait time, at 9.6 days.
* The average wait time for all other categories, including Health & Beauty, Home & Office, Phones & Tablets, and Fashion, is 9.5 days.
* The differences are slight, implying consistent delivery performance across categories.

1. **Recommendation:**

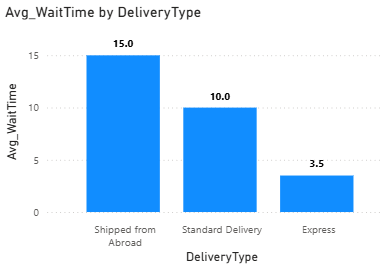
* Maintain consistency in wait times, as they are nearly identical.
* Find out why Electronics is slightly higher (logistics partner, stock availability, or shipping zone concentration).
* Use consistent performance as a selling point in marketing communications.

1. **Explore if there is any relationship between the Delivery type and waiting time between ordering and receiving an item.**
3. **Approach:**

To look into the relationship between delivery type and wait time, I first added a calculated column DeliveryDays to the Orders table, which represents the number of days between OrderDate and DeliveryDate. Then I created a DAX measure called Avg\_WaitTime using: Avg\_WaitTime = Average(Orders[DeliveryDays]). I used a clustered column chart with DeliveryType on the X-axis and Avg\_WaitTime on the Y-axis to compare average wait times for each delivery method.

1. **Work:**

* Calculated DeliveryDays as the difference between delivery and order dates.
* Created a DAX measure to compute the average delivery time.
* Built a visual showing the average wait time for each delivery type: Shipped from Abroad, Standard Delivery, and Express.

****

1. **Insights:**

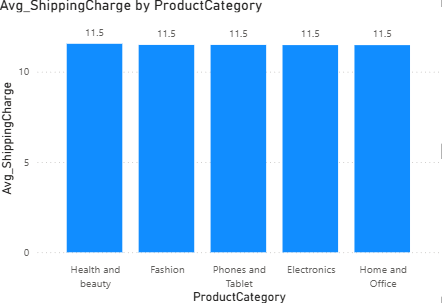
* Orders shipped from abroad have the longest average wait time (15 days), which is most likely due to international shipping and customs clearance.
* Standard Delivery takes approximately 10 days on average, which represents the typical domestic delivery time.
* Express Delivery is much faster, averaging only 3.5 days.
* The method's promised delivery speed and the average wait time show a clear inverse relationship.

1. **Recommendation:**

* For time-sensitive products or customers prioritizing fast delivery, **Express Delivery** should be promoted.
* Improve efficiency in **Standard Delivery** by optimizing local distribution networks to reduce the gap with Express delivery.
* Review partnerships and logistics processes for **Shipped from Abroad** to cut delays and improve customer satisfaction.

1. **Is there any relationship between shipping charges and product type?**
3. **Approach:**

* Calculated the average shipping charge for each product category to check for variations.
* Created a measure: Avg\_ShippingCharge = AVERAGE(Orders[ShippingFee])
* Used a Clustered Column Chart with:X-axis → ProductCategory, Y-axis → Avg\_ShippingCharge

****

1. **Work:**



1. **Insights:**

* The average shipping charge for all product categories is identical at around 11.5.
* This means product type does not influence shipping fee in this dataset.

1. **Recommendations:**

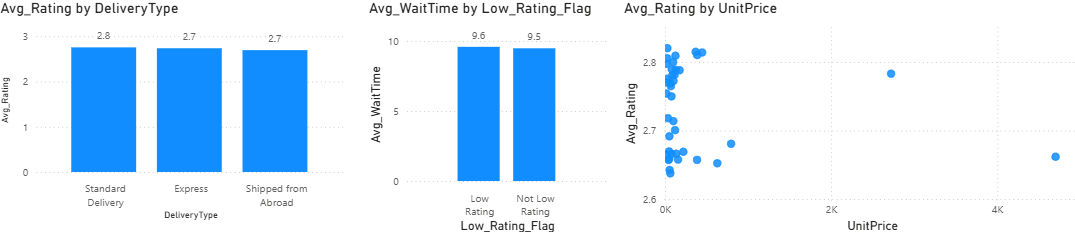
* Since shipping charges are uniform, differentiation based on **weight, size, or delivery urgency** could be explored for more accurate cost allocation.
* If the goal is competitive pricing, the business could **reduce or waive shipping fees** for high-demand categories to attract more customers.

1. **Come up with strategies to decrease the low rating orders after analyzing different factors like waiting time, shipping type, unit price, etc.**
3. **Approach:**

The goal was to identify factors that contribute to low-rated orders by investigating the relationships between waiting time, delivery type, and unit price with customer ratings. Visual analysis and aggregated measures were used to identify patterns and correlations.

1. **Work**:

* Created a bar chart comparing Avg\_WaitTime for Low Rating vs Not Low Rating orders.
* Built a bar chart showing Avg\_Rating by Delivery Type.
* Developed a scatter plot mapping Unit Price against Avg\_Rating to detect price–rating trends.

****

1. **Insights**:

* Waiting Time: Low-rated orders had a slightly longer average wait time (9.6 days) than non-low-rated orders (9.5 days). Although the difference is small, longer wait times can have a negative impact on satisfaction.
* Ratings were similar across all delivery types, including Standard Delivery (2.8), Express (2.7), and Shipped from Abroad (2.7). Expectations set by delivery labels may still influence ratings.
* Unit Price: There was no significant correlation found between price and rating, but some high-priced products received noticeably low ratings, indicating potential quality or expectation gaps.

1. **Recommendations**:

* Faster Delivery & Support: Reduce wait times, give accurate ETAs, and follow up on low ratings with quick resolutions.
* Better Quality & Trust: Ensure quality checks, accurate info, verified reviews, and reward positive feedback.

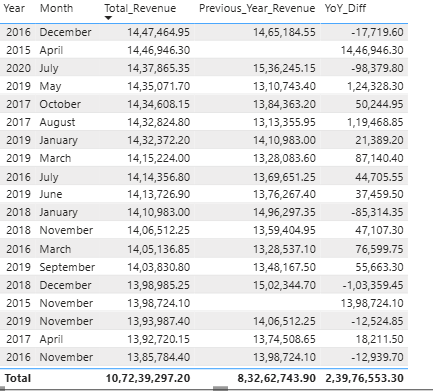
1. **Using the time intelligence DAX function, create a table to compare each month’s sales with the previous year’s same month’s total sales. So, there will be four columns in the output year, month, total sales, previous\_years\_sales.**
3. **Approach:**

* Built a proper date dimension (DateTableQ14) with daily rows and related it to Orders[OrderDate] (Many-to-one, Single direction).
* Marked it as a Date table so time-intelligence works.
* Used measures with time intelligence to compare each month’s sales to the same month last year.

1. **Work**:

****

****

****

* Table visual with columns: Year, Month, Total\_Revenue, Previous\_Year\_Revenue, YoY\_Diff.
* Sorted the table by Total\_Revenue (Descending) to surface the highest months first.

1. **Insights**:

* Several months show a significant YoY increase (tens of thousands), while a few months show sharp declines (e.g., early-2018, mid-2020), indicating inconsistent month-to-month performance.
* The highest-revenue months are spread out over multiple years and months, indicating campaign/event-driven spikes rather than a single, predictable season.
* Months in the first available year have no prior-year comparator (PY blanks) and should be treated as the baseline rather than under/overperformance.
* A small number of months account for a disproportionate share of YoY growth, implying that repeatable tactics are worth isolating and scaling.

1. **Recommendations**:

* Look for months with significant year-over-year gains and repeat the same promotions, prices, and product bundles in those months the next year.
* Check which products or regions fell behind last year and offer targeted discounts or visibility boosts there.
* Stock up and schedule extra staff for historically strong months; confirm supplier capacity 4-6 weeks ahead of those peak periods.
* Add a monthly "This year vs last year" percentage to the dashboard; if a month falls below target, take immediate action (promotion, price tweak, or stock shift).

1. **What do you understand by Power BI gateway? What are its use cases?**

* A secure bridge between on-premises data sources and the cloud-based Power BI service.
* Used when using a local dataset but want to refresh/update reports on the Power BI Service without manually uploading data.
* Use cases:

1. Automates secure data refresh from on-premise sources like SQL Server, Excel, or ERP/CRM systems to keep Power BI dashboards up-to-date without manual uploads.
2. Enables real-time insights by connecting local business systems to the cloud while maintaining strict data security and compliance.
3. **How would you approach this problem, if the objective and subjective questions weren't given?**

* First, I would conduct a thorough data review going over all tables, relationships, and any missing or inconsistent data.
* Then I'd define the key KPIs that are relevant to the business goals, such as revenue growth, delivery times, and repeat purchases.
* Next, I'd conduct exploratory analysis to identify trends, seasonal patterns, and unusual spikes or drops in performance.
* I would prioritize the insights based on their potential business impact, ensuring that high-value actions are addressed first.
* After that, I'd create interactive dashboards so that stakeholders could easily explore the data and find answers for themselves.
* Finally, I would implement a continuous review process to ensure that the analysis remains accurate, relevant, and up to date as the business evolves.