Contents

Figure Table

Execution Summary

The project report on **E-Commerce Jewelry Store** provides comprehensive information on a number of topics, with an emphasis on a dataset from an online jewelry store. **The order datetime, order ID, purchased product ID, quantity of SKU in the order, category ID, category alias, brand ID, price in USD, user ID, product gender (separated into male and female categories), metal, and stone** are among the crucial fields included in the dataset.

With each transaction, the **"Order datetime"** gives a timestamp that sheds light on historical trends and patterns in purchases. Every order has a unique **"Order ID"** that connects it to other pertinent information. While **"Quantity of SKU in the order"** specifies the quantity of units purchased, "Purchased product ID" refers to the exact jewelry item bought in a transaction. In addition to offering details on the kind of jewelry being sold, the **"Category ID"** and **"Category alias"** also provide an analytical framework that is hierarchical. **"Brand ID"** helps with brand-centric insights by identifying the brand linked to each product. For financial analysis, **"Price in USD"** is essential since it clarifies client spending trends and revenue generation. Each customer's **"User ID"** acts as a distinct identifier, making it possible to trace their specific purchase patterns. Items are categorized according to gender in the "Product gender" field, making gender-specific analysis possible. The qualities "metal" and "stone" furnish information regarding the jewelry's composition, facilitating an examination of buyers' material preferences.

This extensive dataset of Jewelry store gives analysts and researchers the knowledge they need to explore many facets of the online jewelry store industry. The dataset offers a comprehensive foundation for intelligent investigation and calculated decision-making in the field of e-commerce jewelry retail, from comprehending consumer preferences and spending habits to assessing product popularity and brand effectiveness.

Introduction

The e-commerce jewelry store dataset project report provides an in-depth examination of the complex realm of online jewelry retail. It does this by utilizing a

rich dataset that includes vital details like the **product's gender (male or female), metal, and stone, quantity of SKUs purchased, category ID, category alias, brand ID, price in USD, user ID, and order date/time.** This comprehensive dataset gives a detailed view of the dynamics inside the e-commerce jewelry industry by capturing the core of client transactions. The report's analysis of these data sets attempts to reveal trends in customer behavior, preferences for particular product features, and the interactions between elements like cost, brand affinity, and material selections. These kinds of insights are essential for e-commerce companies who want to keep on top of the ever-changing trends in the exciting world of online jewelry shopping, optimize their strategy, and improve user experiences. We hope that this project report will give stakeholders insightful information on the workings of an online jewelry store, empowering them to make better decisions about how to increase customer satisfaction, maximize inventory, and boost overall business performance.
Basically, the main target of this project report is to present a thorough study of a dataset from an online jewelry retailer, illuminating a number of factors such order trends, product preferences, and consumer behavior.

Data Source Description

Link :

https://www.kaggle.com/datasets/mkechinov/ecommerce-purchase-history-from-jewelry-store

The order **datetime, order ID, purchased product ID, quantity of SKU in the order, category ID, category alias, brand ID, price in USD, user ID, product gender (indicating whether the product is intended for males or females), metal composition, and stone** details are among the crucial details for every transaction that make up the e-commerce jewelry store dataset. Together, these characteristics offer a thorough understanding of the sales dynamics of the store, allowing for analysis of factors like as order trends, product popularity, pricing schemes, and user preferences with regard to gender, metal type, and stone preference. In this dataset numerous components can be optimized and the complex factors impacting consumer behavior can be understood. The below data sheet has around 95912 rows and 12 columns.

E-Commerce Jewelry Store dataset explanation Table

ID	COLUMN NAME	DESCRIPTION	DATA TYPE
1	Order datetime	Shows order date and time	DATE/TIME

2	Order ID	Shows order id	INT
3	Purchased product ID	Shows purchased product id	INT
4	Quantity of SKU in the order	Quantity of sku in order	INT
5	Category ID	Unique id of category	INT
6	Category alias	Category alias	VAR/CHAR
7	Brand ID	Unique brand id	INT
8	Price in USD	Price in us dollar	DECIMAL
9	User ID	User id	VAR/CHAR
10	Product gender (for male/female)	Product for male or female	VAR/CHAR
11	Metal	Metal name	VAR/CHAR
12	Stone	Stone used	DATE/TIME

Related Business Questions

- In which month and year, the maximum amount is generated?
- Which product generated the highest total revenue?
- Which stone's products were sold the most?
- In which time period the highest number of orders were received?
- What was the total targeted revenue?
- How many types of colors of jewelry are available in jewelry stores?
- Which color of jewelry was liked the most due to which its sale was the highest?

Report Objectives

- **Analysis of Customer Behavior -** Examine and evaluate consumer behavior trends found in the data of online jewelry store. Determine patterns in order frequency, purchase timing, and relationships with user demographics (age, gender, etc.). Examine how different dataset factors affect consumer engagement and purchasing preferences.
- **Preferences for Product Features -** Look through the dataset to find common product attributes like stone and metal kinds and examine the relationships between product characteristics (stone, metal) and consumer choices. After that

find out if there are any particular feature combinations that the buyers find more appealing.

- **Brand Loyalty and Affinity -** Examine the dataset to determine brand affinity and customer loyalty and how brand identification affects consumer choice and loyalty.
- **Price Sensitivity and Trends in the Market -** Examine how pricing affects consumer behavior when making purchases and how changes in price affect sales over time. Determine pricing trends in the market and the effects they will have on the online jewelry business.
- **Strategies for Business Performance Optimization -** On the basis of dataset insights, suggest optimization solutions for the online jewelry retailer and make suggestions on product assortment, pricing tactics, and inventory control. Give recommendations for improving the user experience by utilizing the patterns and preferences that have been found.

Key Findings

This section will examine the work done on Power Bi and the dataset visualizations in order to, solve the business problems and provide relevant results and recommendations.

Below dashboard shows the overall representation of available data in which total amount generated, total customers and total delivered orders are shown by the help of number of graphs

Amount wise findings

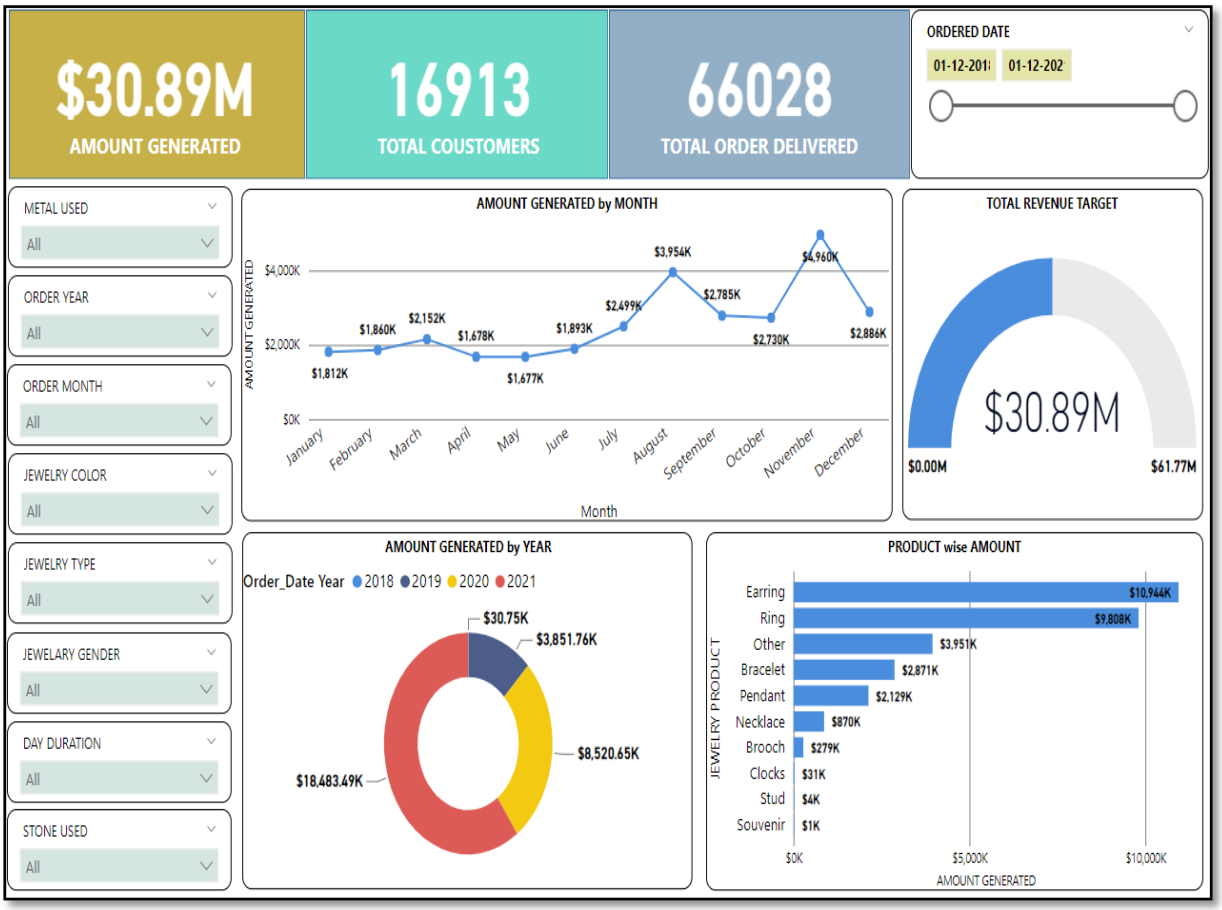


Figure-1: Final View of dashboard JEWELRY_AMOUNT

Month wise Generated Amount:

In the below chart amount generated is shown according to month. In the Month of November, the generated amount is highest as compare to other months.

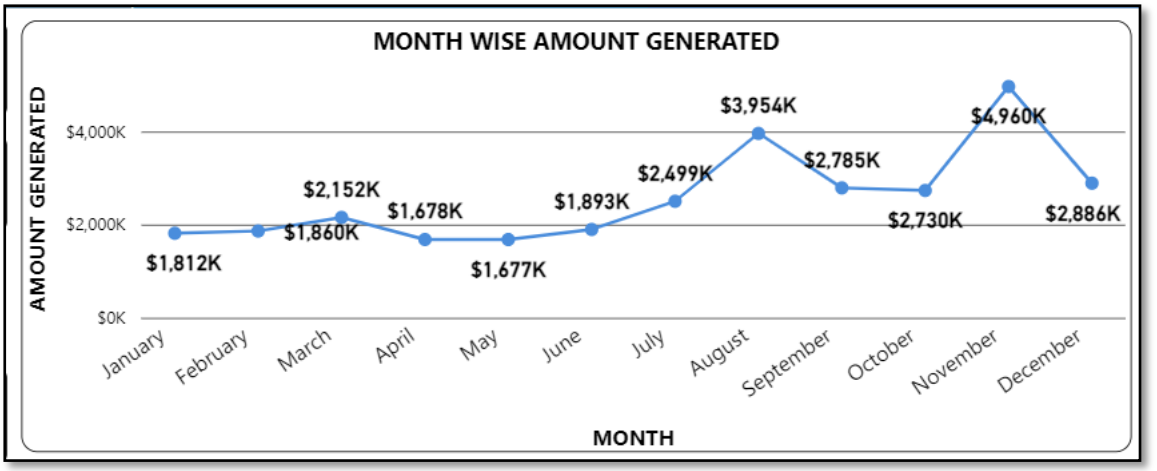


Figure-2: Graph- Month wise Generated Amount

Year wise Generated Amount:

Below donut chart shows the total amount generated with year. According to below representation in 2021 the highest amount was generated shown by orange color in the figure

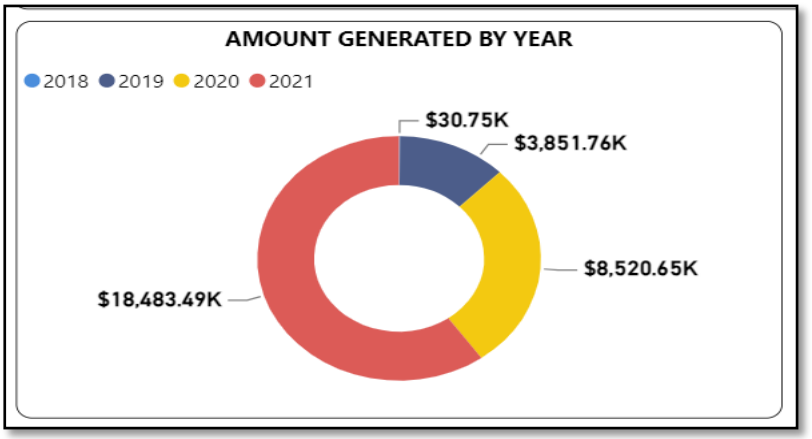


Figure-3: Graph- Year wise Generated Amount

Product Items wise Amount:

In the below field amount generated by product was shown with the help of chart. According to this highest amount was generated from ring.

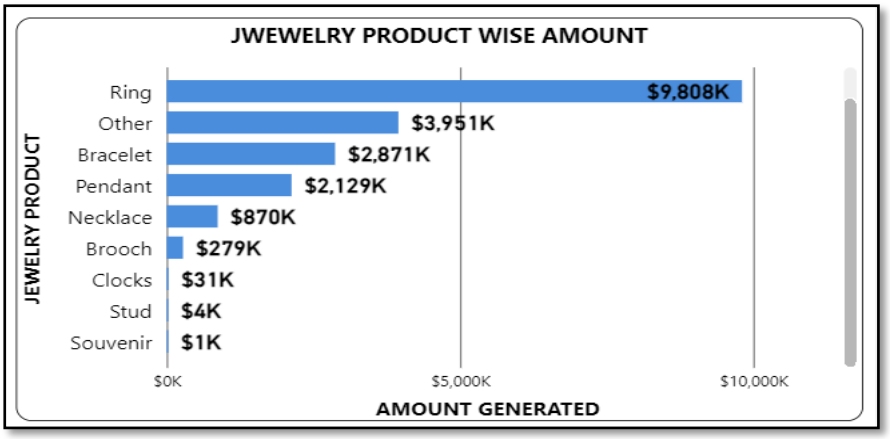


Figure-4: Graph- Product Items wise Amount

Order Wise conclusion

Below dashboard shows the total amount generated with total quantity sold -

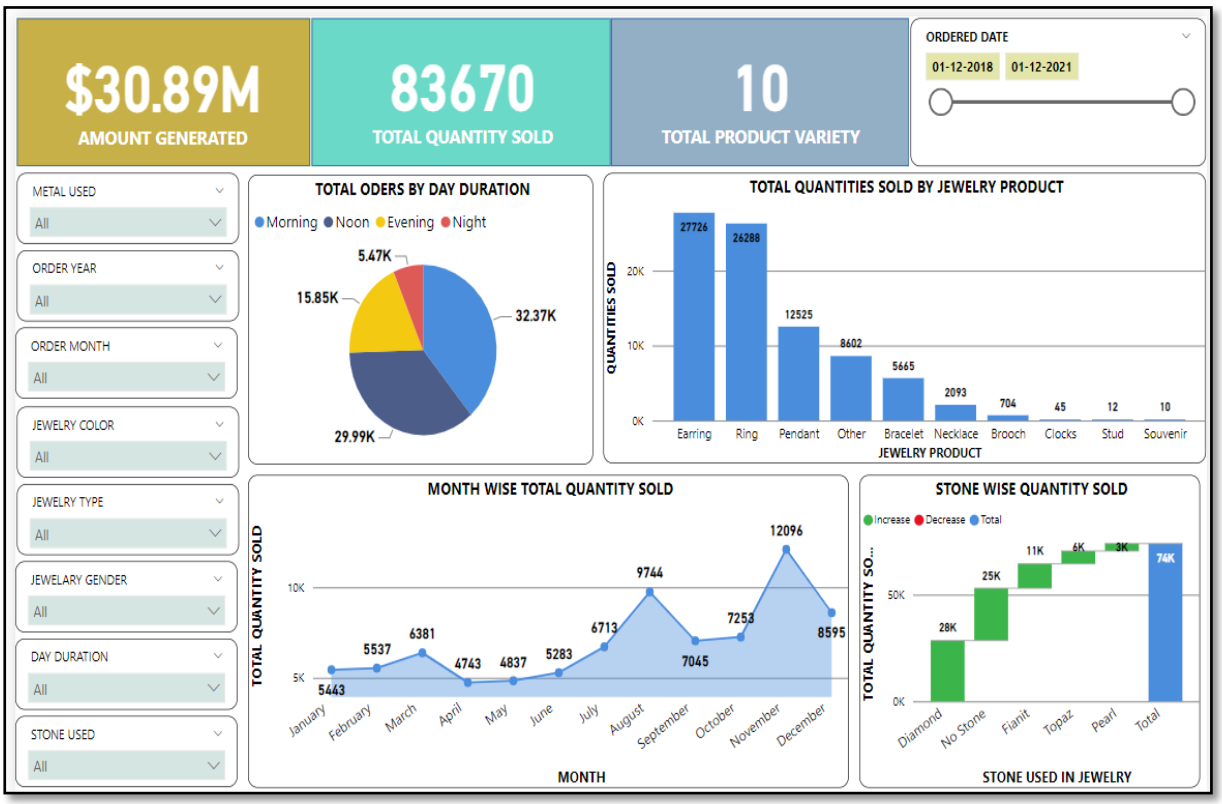


Figure-5: Final View of Dashboard JEWELRY_ORDER

Total Quantity sold by Jewelry product:

Below chart shows total quantity of products sold by name of product. As per the given representation highest quantity of earrings was sold out

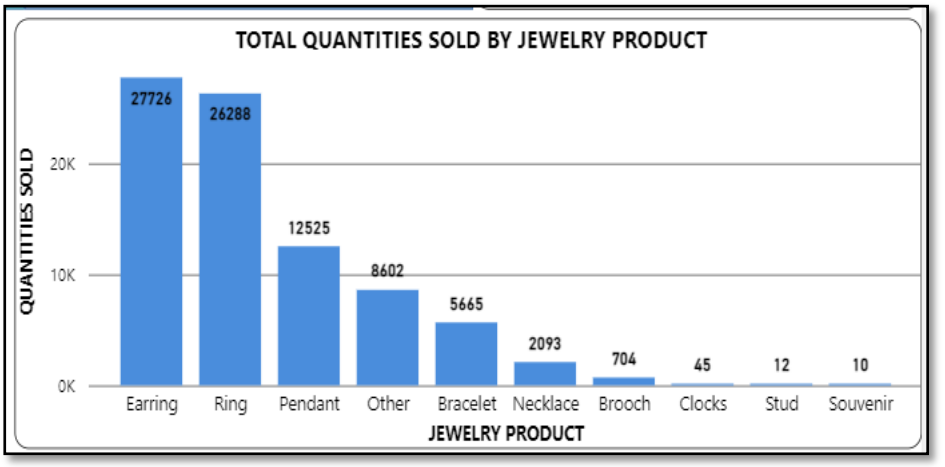


Figure-6: Graph- Total Quantity sold by Jewelry product

Total Orders by Day Time Duration:

According to below representation in the time of morning , highest orders were placed and the number of orders is around 32.37K

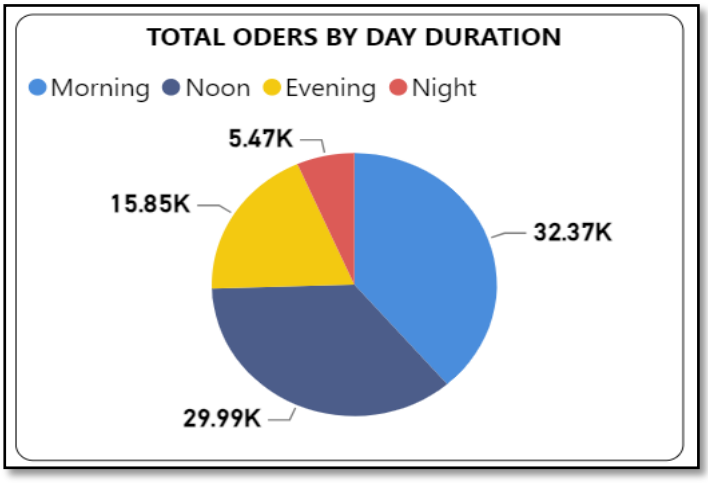


Figure-7: Graph- Total Orders by Day Time Duration

Month Wise Total Quantities Sold:

Below chart shows the total quantity of product sold according to the month

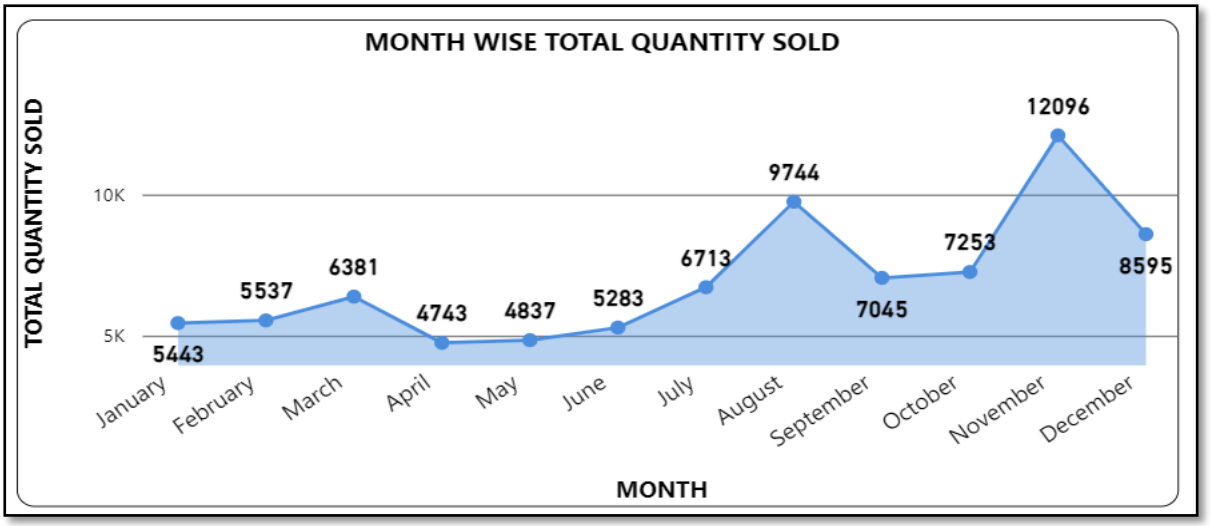


Figure-8: Graph- Month Wise Total Quantities Sold

Jewelry Stone wise Quantities Sold:

Given chart shows total quantity sold of particular stone product. The graph also shows the total amount of each type of stone that was used in jewelry in the past year. Diamonds were the most popular stone, with 28K stones used. No stone jewelry was the second most popular, with 25K pieces sold. Fianit was the third most popular stone, with 11K stones used.

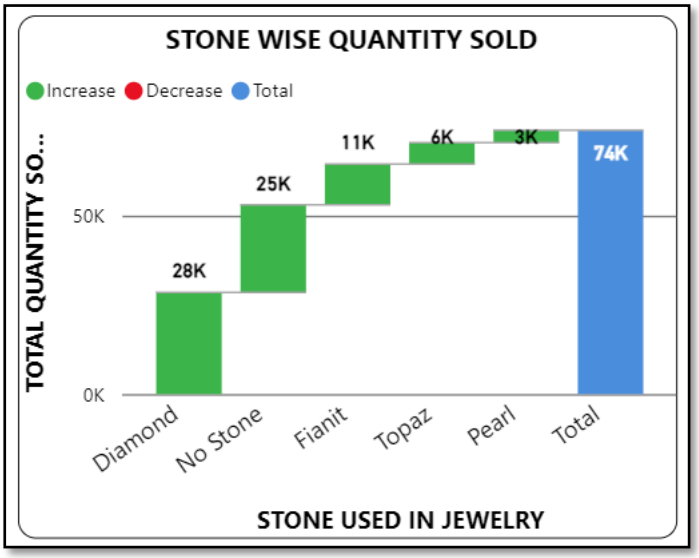


Figure-9: Graph- Jewelry Stone wise Quantities Sold

Product orders wise summary

Below dashboard shows the total brands available in the store with total product varieties. According to this representation total brands available in the store is 6 and the variety of products is 10.

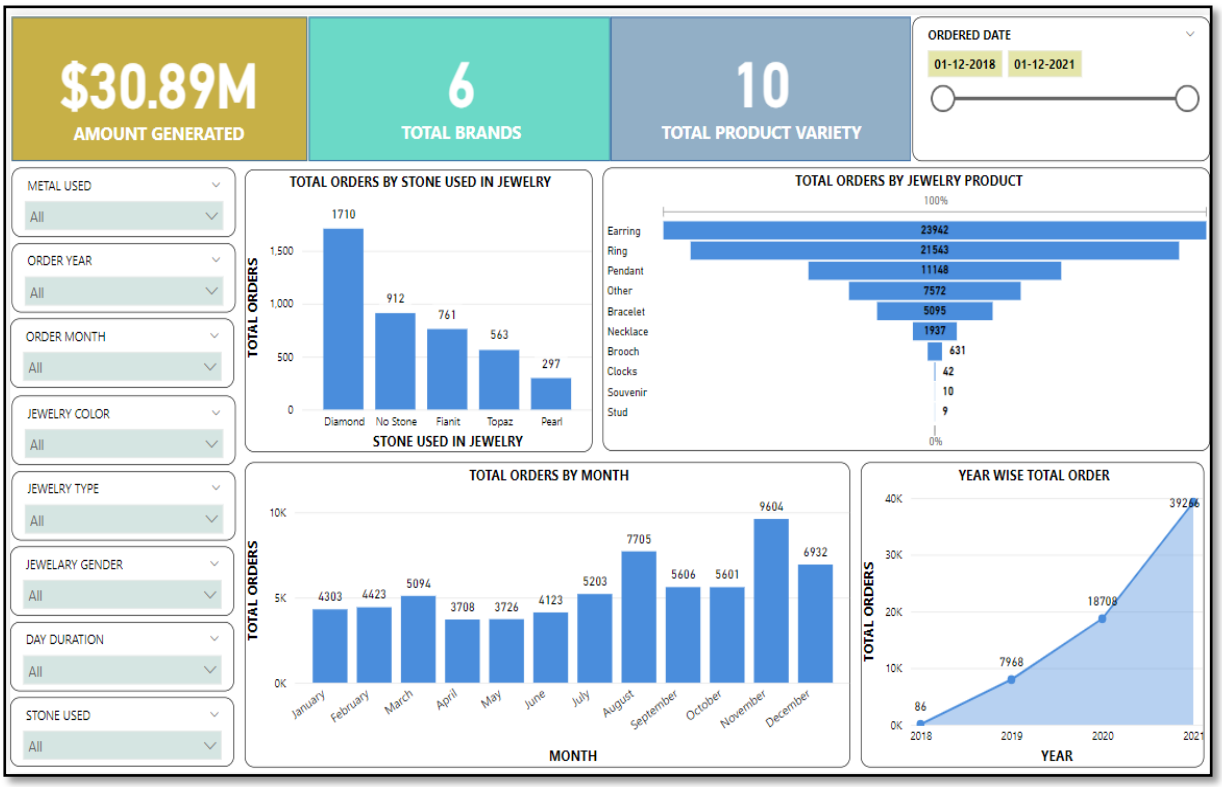


Figure-10: Final View of dashboard JEWELRY_CUSTOMER

Total Orders by Jewelry Stone:

The graph shows that diamonds are the most popular stone, with 1,710 orders placed. There were 912 orders for jewelry with no stone, 761 orders for jewelry with fianit, 563 orders for topaz jewelry, and 297 orders for pearl jewelry

Figure-11: Graph- Total Orders by Jewelry Stone

Total Orders by Jewelry Product:

Below chart shows the total ordered product by category of product. According to these earnings sold out the most as compared to other products.

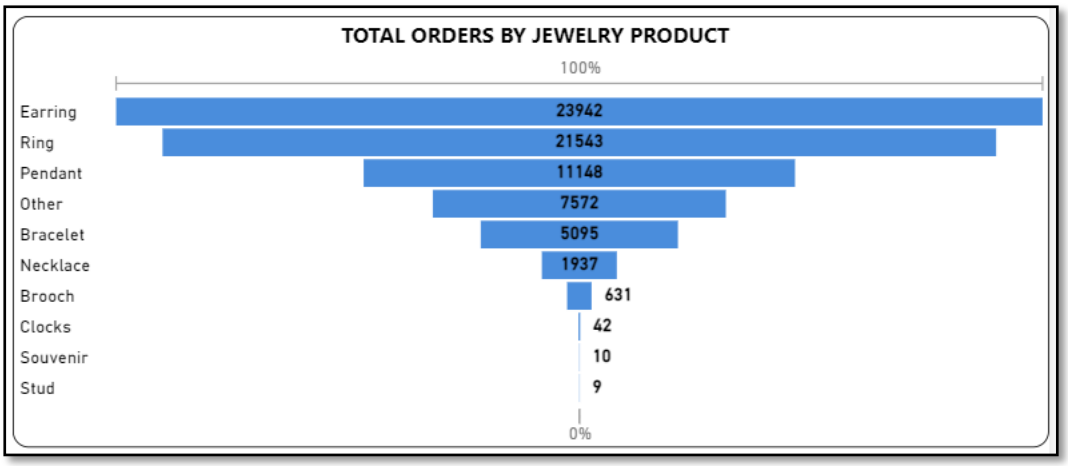


Figure-12: Graph- Total Orders by Jewelry Product

Total Order by Month:

It shows a graph of the number of orders placed for jewelry by month, with each data point labeled with the total number of orders for that month.

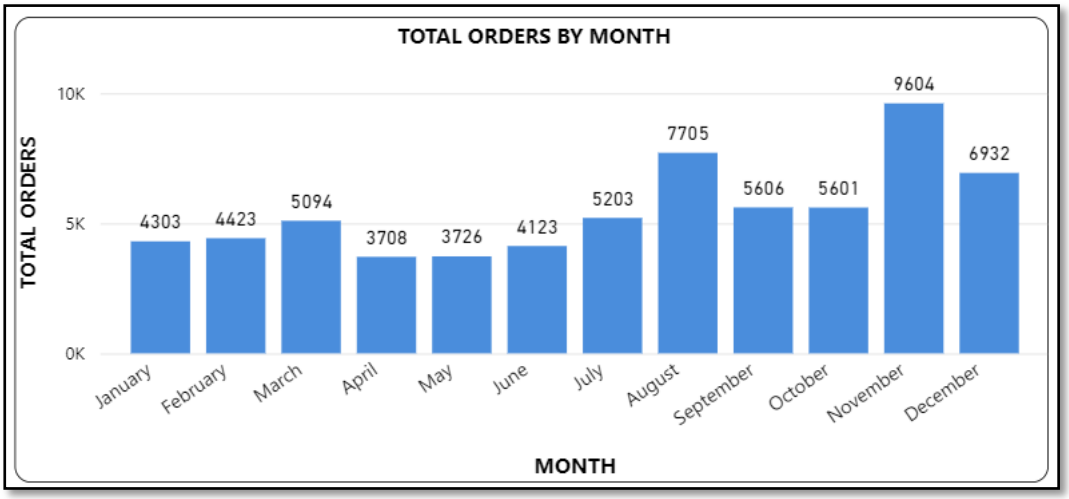


Figure-13: Graph- Total Order by Month

Year Wise Total Orders:

It appears to be a line graph showing total orders placed over a year, likely for a business.

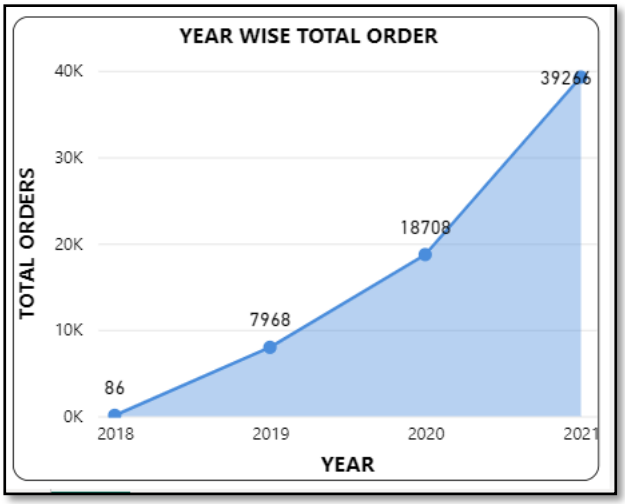


Figure-14: Graph- Year Wise Total Orders

Jewelry Analysis summary

In the below dashboard the summarize representation of amount generated with total number of customers and total delivered orders are shown -

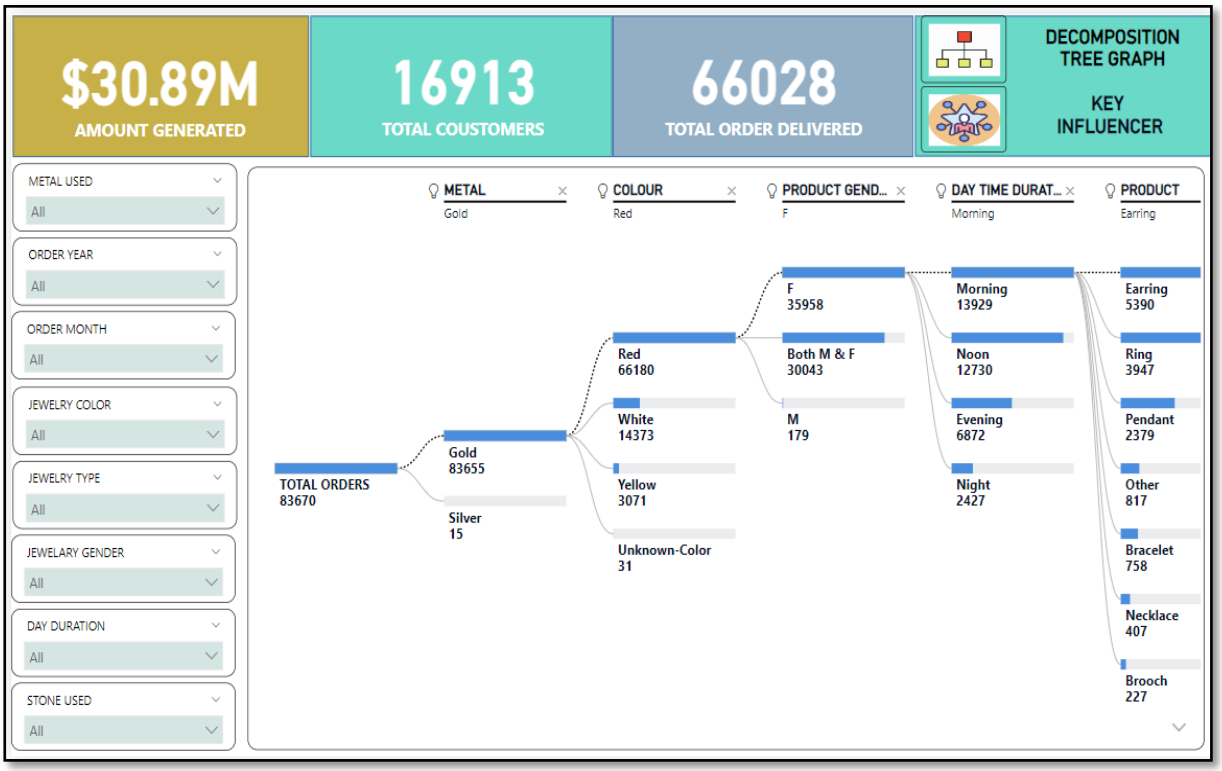


Figure-15: Final View of dashboard JEWELRY_ANALYSIS

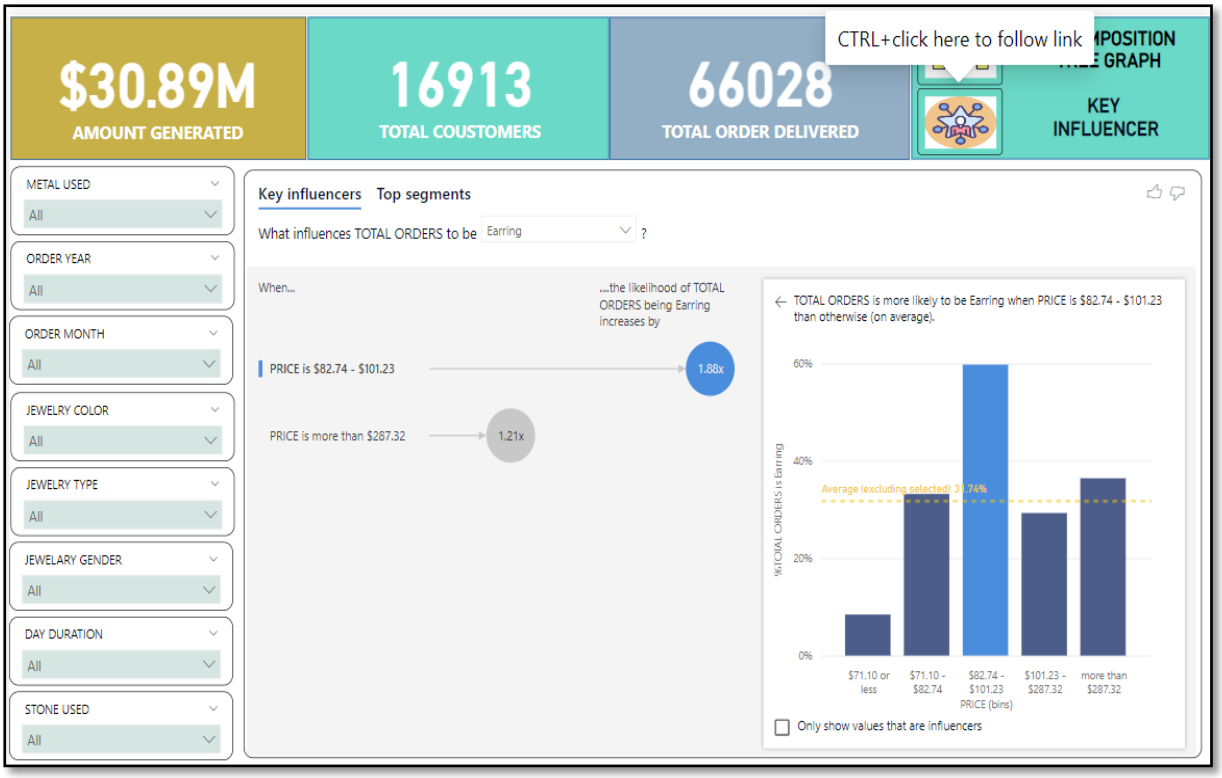


Figure-16: Final View of dashboard JEWELRY_ANALYSIS

Question-Answers

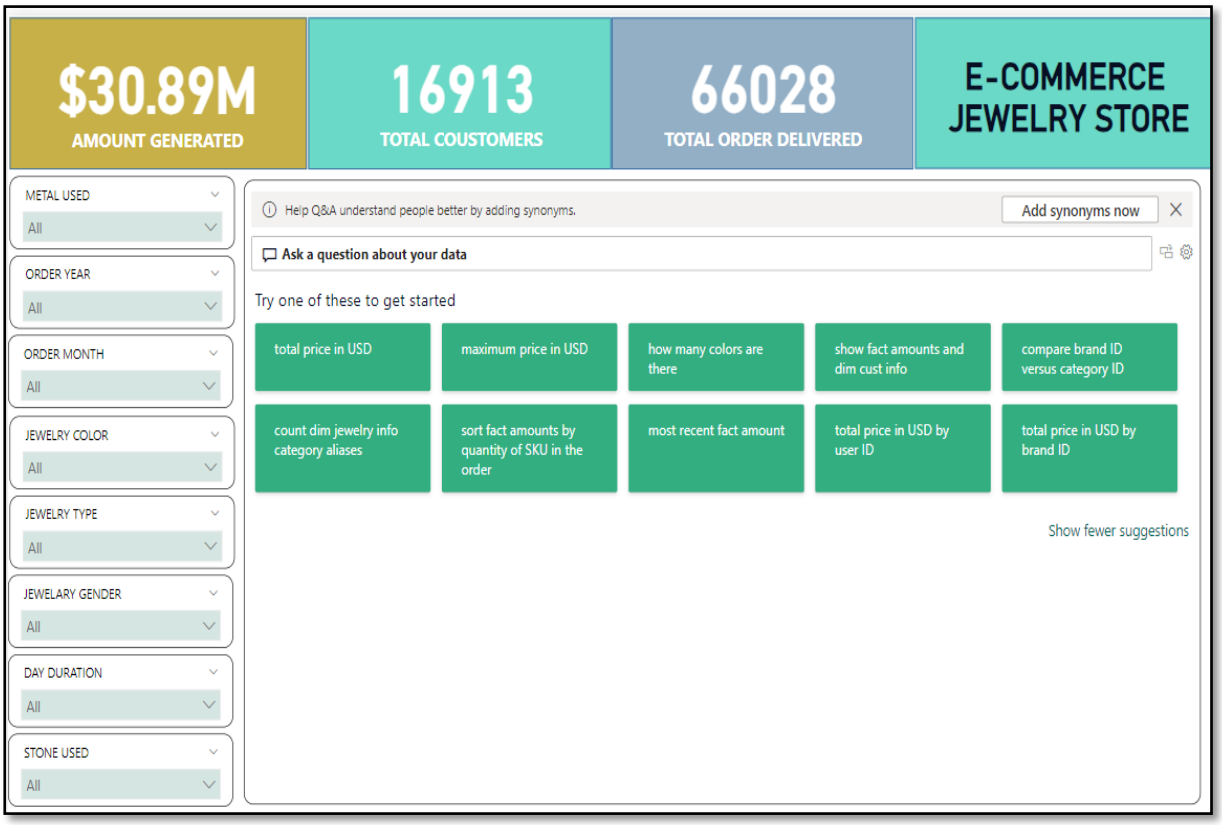


Figure-17: Final View of dashboard JEWELRY_QnA

Conclusions

To sum up, the examination and evaluation of the data from the jewelry store retailer have yielded significant understandings into the workings of the online jewelry market. A thorough understanding of customer preferences and purchasing patterns has been obtained through the analysis of important attributes like order datetime, order ID, product details, quantity, category information, brand details, pricing, user demographics, and product characteristics like gender, metal, and stone. After deeply studying this dataset, we have reached some following conclusions :

- Total generated revenue was the highest in the year 2020
- The sales of earrings were the highest due to which it received more revenue.
- Diamond products were sold the most.

Recommendations

- The company should focus on increasing the sales of its pearl products.
- The company should focus on improving the sales of its other products like earrings.

References

1. Fedirko, O., Zatonatska, T., Wołowiec, T. and Skowron, S., 2021. Data science and marketing in E-commerce amid COVID-19 pandemic.

2. Li, P., 2016. Current and future years of e-commerce. In Encyclopedia of e-commerce development, implementation, and management (pp. 1031-1044). IGI Global.

3. Jurowski, K., 2023. The toxicological assessment of hazardous elements (Pb, Cd and Hg) in low-cost jewelry for adults from Chinese E-commerce platforms: In situ analysis by portable X-ray fluorescence measurement. Journal of Hazardous Materials, 460, p.132167.

4. Cheng, V.W., Choi, S.H. and Cheung, H.H., 2019. Mixed Reality for Virtual Jewellery Fitting. Journal of Marketing Development and Competitiveness, 13(3), pp.25-37.

5. Aspin, A., 2014. High Impact Data Visualization with Power View, Power Map, and Power BI. Apress.

Appendix

Data must be converted or made ready for use in the future by organizing, merging, and structuring it for use in analytics, business intelligence, and visualizations. This will allow for the trustworthy acquisition of insights from the data (Stedman, Burns and Pratt,2022).

Additionally, data preparation can aid in producing reliable and consistent data outputs. As a result, the actions listed below were taken to construct an analysis in Power-BI: locating the data, importing it into Power-BI, processing the data, modeling it, and carrying out the necessary changes before visualizing, designing, and branding the information.

Importing Data-Sheet in power bi desktop

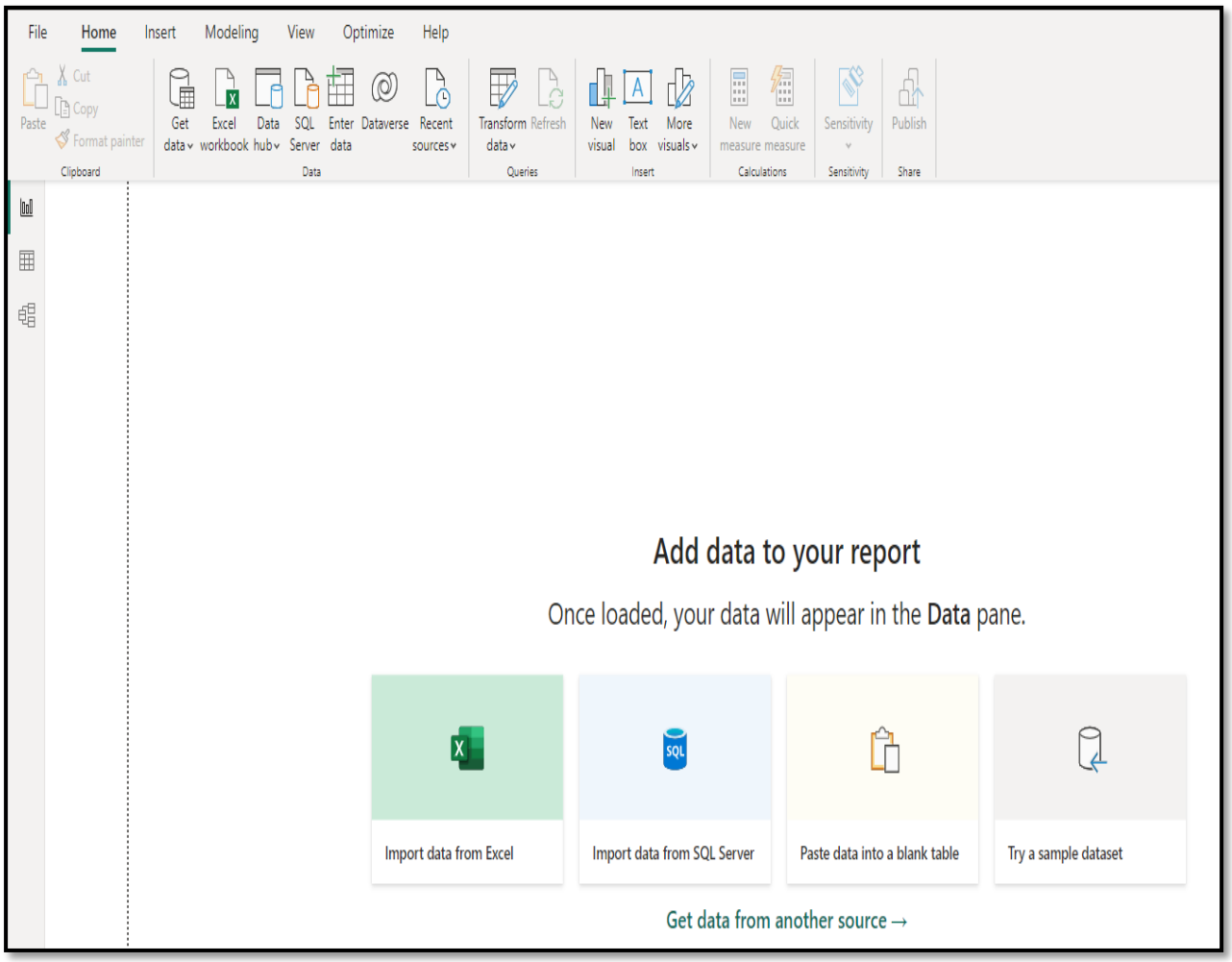


Figure-18: Opening Page of Power-BI

Load the data sheet in power-bi

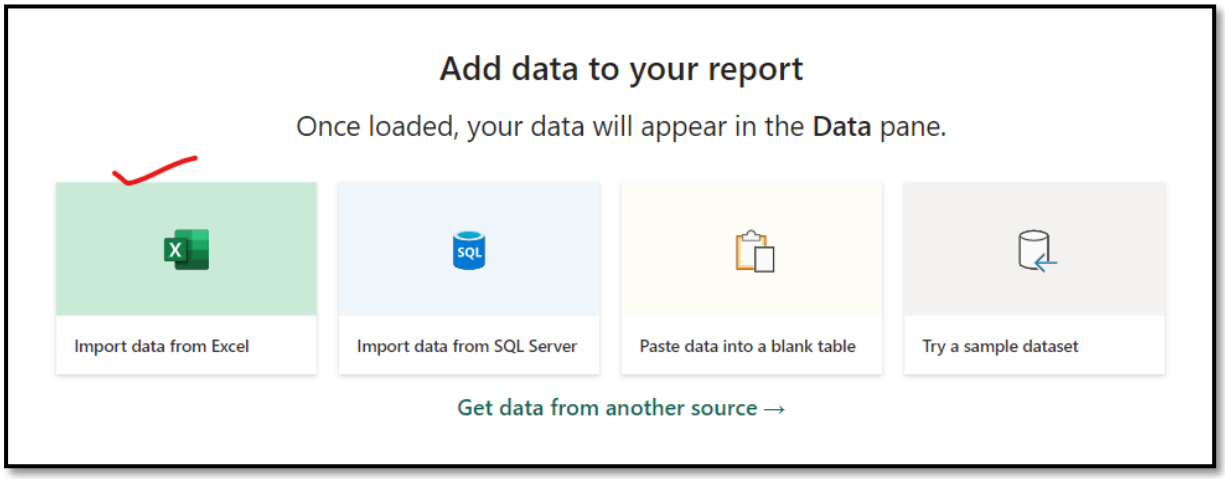
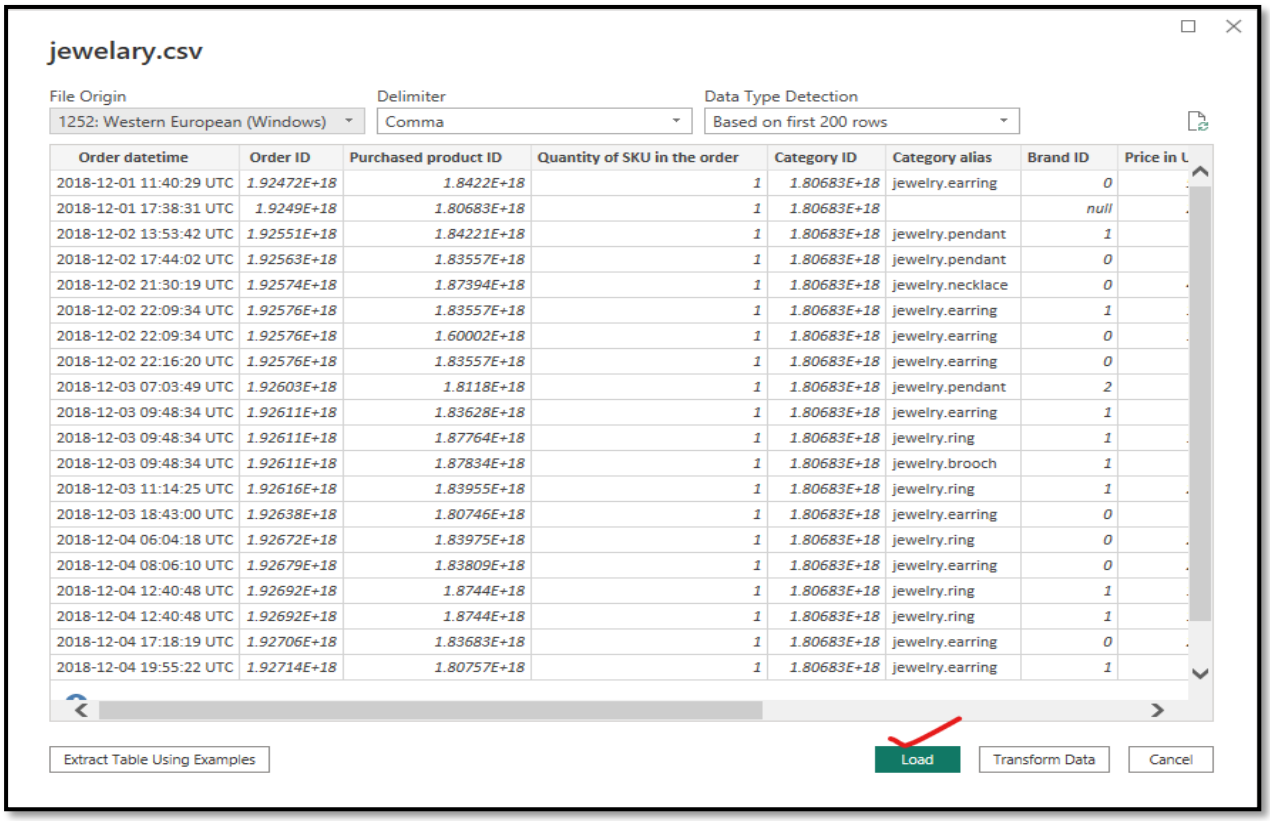


Figure-19: Data Importing Option

- After that, select the file from Excel. And load them.



Order datetime	Order ID	Purchased product ID	Quantity of SKU in the order	Category ID	Category alias	Brand ID	Price in U
2018-12-01 11:40:29 UTC	1.92472E+18	1.8422E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-01 17:38:31 UTC	1.9249E+18	1.80683E+18	1	1.80683E+18		null	
2018-12-02 13:53:42 UTC	1.92551E+18	1.84221E+18	1	1.80683E+18	jewelry.pendant	1	
2018-12-02 17:44:02 UTC	1.92563E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant	0	
2018-12-02 21:30:19 UTC	1.92574E+18	1.87394E+18	1	1.80683E+18	jewelry.necklace	0	
2018-12-02 22:09:34 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring	1	
2018-12-02 22:09:34 UTC	1.92576E+18	1.60002E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-02 22:16:20 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-03 07:03:49 UTC	1.92603E+18	1.8118E+18	1	1.80683E+18	jewelry.pendant	2	
2018-12-03 09:48:34 UTC	1.92611E+18	1.83628E+18	1	1.80683E+18	jewelry.earring	1	
2018-12-03 09:48:34 UTC	1.92611E+18	1.87764E+18	1	1.80683E+18	jewelry.ring	1	
2018-12-03 09:48:34 UTC	1.92611E+18	1.87834E+18	1	1.80683E+18	jewelry.brooch	1	
2018-12-03 11:14:25 UTC	1.92616E+18	1.83955E+18	1	1.80683E+18	jewelry.ring	1	
2018-12-03 18:43:00 UTC	1.92638E+18	1.80746E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-04 06:04:18 UTC	1.92672E+18	1.83975E+18	1	1.80683E+18	jewelry.ring	0	
2018-12-04 08:06:10 UTC	1.92679E+18	1.83809E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1	
2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1	
2018-12-04 17:18:19 UTC	1.92706E+18	1.83683E+18	1	1.80683E+18	jewelry.earring	0	
2018-12-04 19:55:22 UTC	1.92714E+18	1.80757E+18	1	1.80683E+18	jewelry.earring	1	

Figure-20: View of the Data Sheet

Figure-21: pop up of loading data

- There are 21 columns in the data sheet.

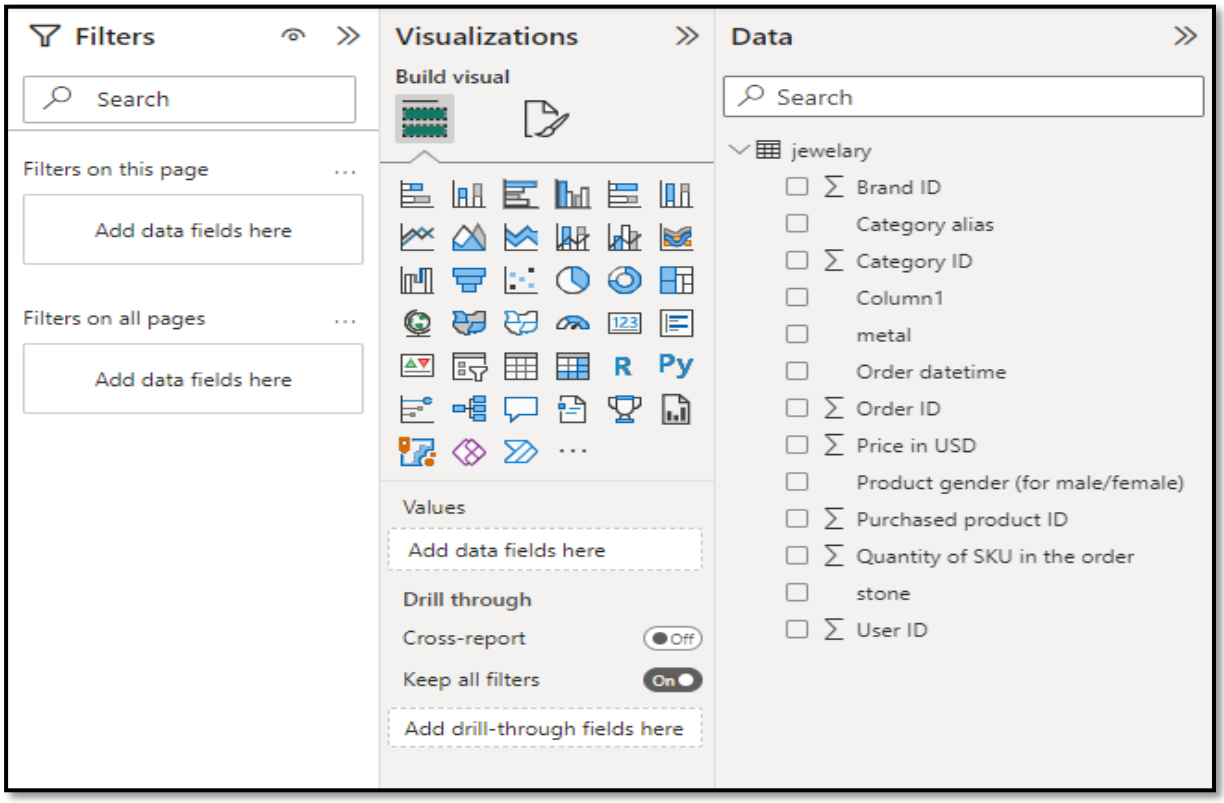


Figure-22: Data View In power BI

Data Cleansing & Transformation Steps in Datasheet

The datasets must be cleaned in order for analysis to take place and for visualisations to be displayed with accuracy and dependability. The convert data button on the home tab was clicked, opening the data cleansing tool.

The data will be preprocessed and cleaned using the power query editor. The query editor has a number of tools for cleaning and transforming data.

In order to transform the data set, click on “Transform data” option available in the ribbon menu.

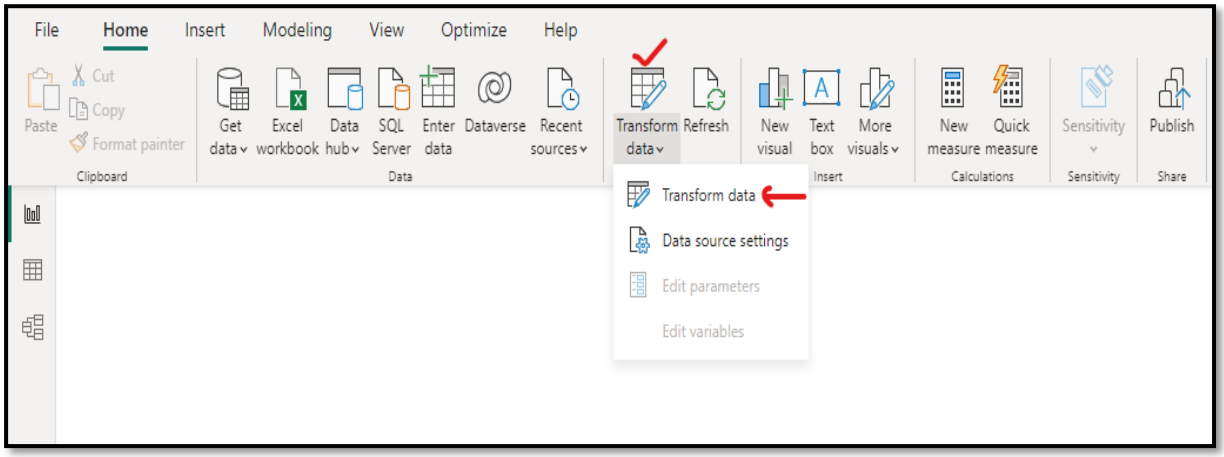
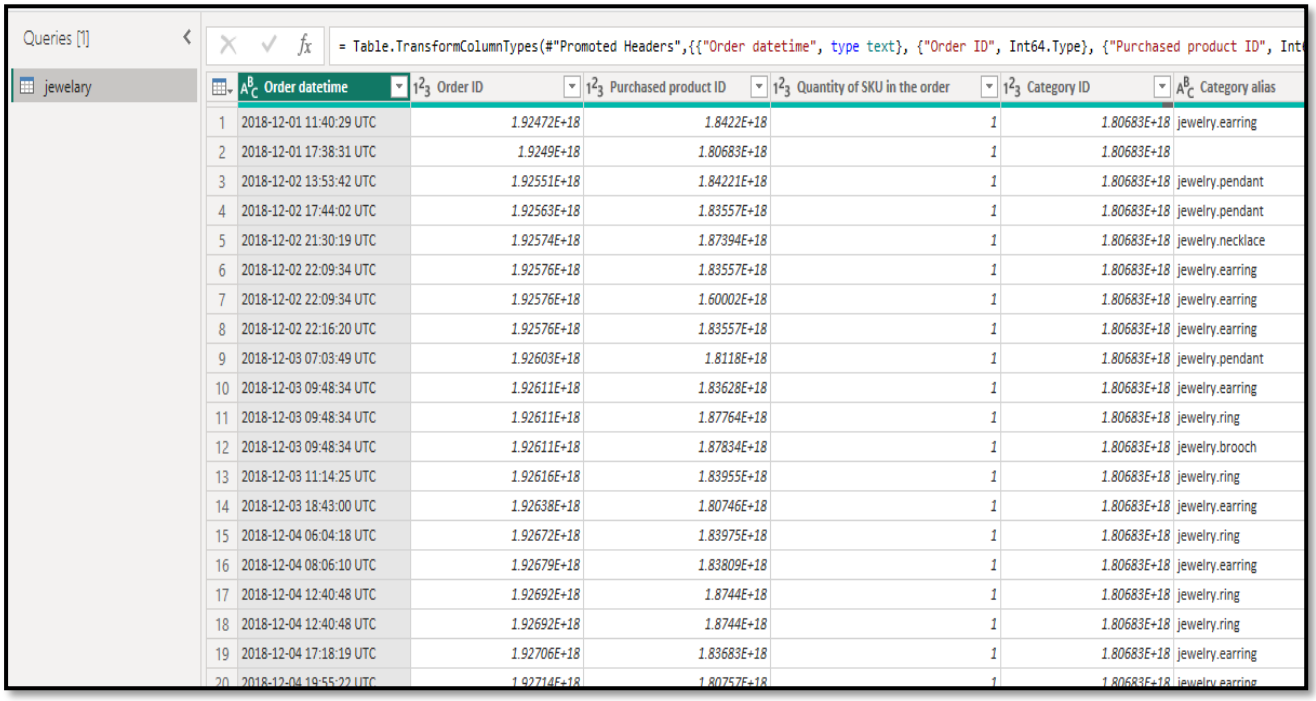


Figure-23: Data Transformation Option



	Order datetime	Order ID	Purchased product ID	Quantity of SKU in the order	Category ID	Category alias
1	2018-12-01 11:40:29 UTC	1.92472E+18	1.8422E+18	1	1.80683E+18	jewelry.earring
2	2018-12-01 17:38:31 UTC	1.9249E+18	1.80683E+18	1	1.80683E+18	
3	2018-12-02 13:53:42 UTC	1.92551E+18	1.84221E+18	1	1.80683E+18	jewelry.pendant
4	2018-12-02 17:44:02 UTC	1.92563E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant
5	2018-12-02 21:30:19 UTC	1.92574E+18	1.87394E+18	1	1.80683E+18	jewelry.necklace
6	2018-12-02 22:09:34 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring
7	2018-12-02 22:09:34 UTC	1.92576E+18	1.60002E+18	1	1.80683E+18	jewelry.earring
8	2018-12-02 22:16:20 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring
9	2018-12-03 07:03:49 UTC	1.92603E+18	1.8118E+18	1	1.80683E+18	jewelry.pendant
10	2018-12-03 09:48:34 UTC	1.92611E+18	1.83628E+18	1	1.80683E+18	jewelry.earring
11	2018-12-03 09:48:34 UTC	1.92611E+18	1.87764E+18	1	1.80683E+18	jewelry.ring
12	2018-12-03 09:48:34 UTC	1.92611E+18	1.87834E+18	1	1.80683E+18	jewelry.brooch
13	2018-12-03 11:14:25 UTC	1.92616E+18	1.83955E+18	1	1.80683E+18	jewelry.ring
14	2018-12-03 18:43:00 UTC	1.92638E+18	1.80746E+18	1	1.80683E+18	jewelry.earring
15	2018-12-04 06:04:18 UTC	1.92672E+18	1.83975E+18	1	1.80683E+18	jewelry.ring
16	2018-12-04 08:06:10 UTC	1.92679E+18	1.83809E+18	1	1.80683E+18	jewelry.earring
17	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring
18	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring
19	2018-12-04 17:18:19 UTC	1.92706E+18	1.83683E+18	1	1.80683E+18	jewelry.earring
20	2018-12-04 19:55:22 UTC	1.92714E+18	1.80757E+18	1	1.80683E+18	jewelry.earring

Figure-24: View of Data Transformation

= Table.TransformColumnTypes(#"Promoted Headers",{{"Order datetime", type text}, {"Order ID", Int64.Type}, {"Purchased product ID", Int64.Type}, {"Quantity of SKU in the order", Int64.Type}, {"Category ID", Int64.Type}, {"Category alias", type

text}, {"Brand ID", Int64.Type}, {"Price in USD", type number}, {"User ID", Int64.Type}, {"Product gender (for male/female)", type text}, {"colour", type text}, {"metal", type text}, {"stone", type text}})

Removing Null Columns from Data Sheet

If the datasheet has any column completely null then it can be removed from the data sheet.

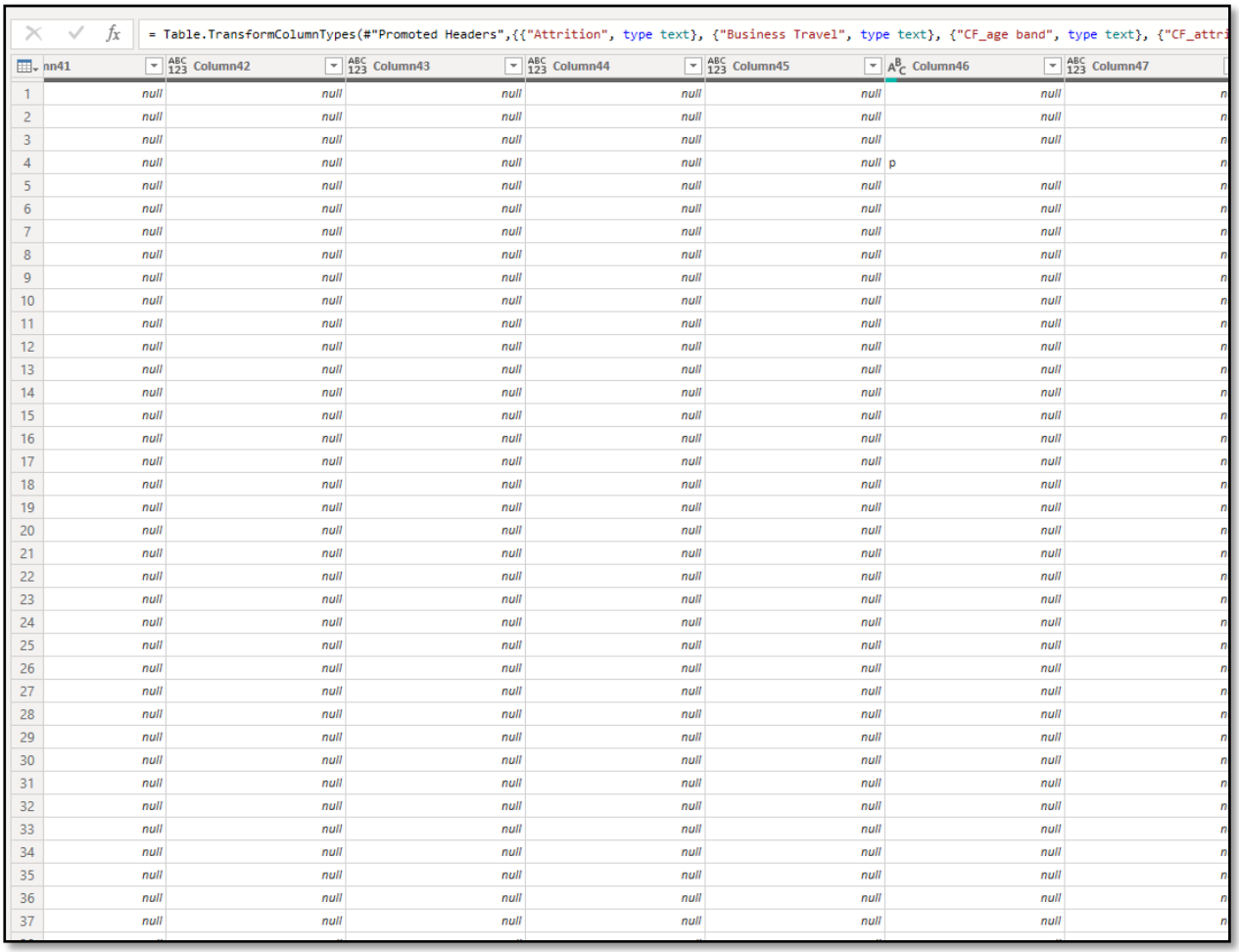


Figure-25: Null Columns in Dataset

The null column can removed by two ways, Either by options “remove Columns” in Ribbon tab or By clicking right column on null column header.

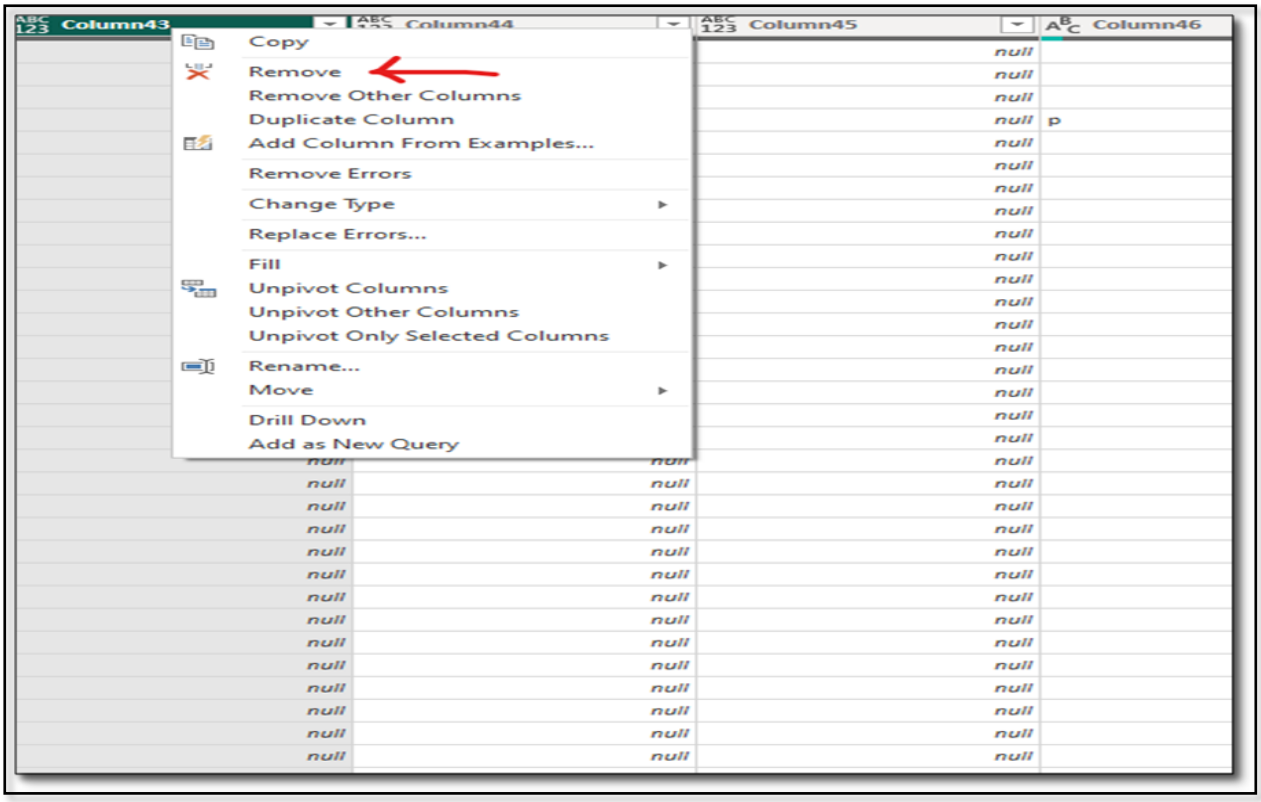


Figure-26: Removing Null Columns

Removing Null values in Column

Delete null values from your Power BI data sets to prevent problems. You may make sure that your analyses and visualizations are based on valid and full data by eliminating null values. Removing null values can also speed up the processing of your Power BI dashboards and reports because it will require less data to analyze. If any column is having null then it can be replaced using replace values.

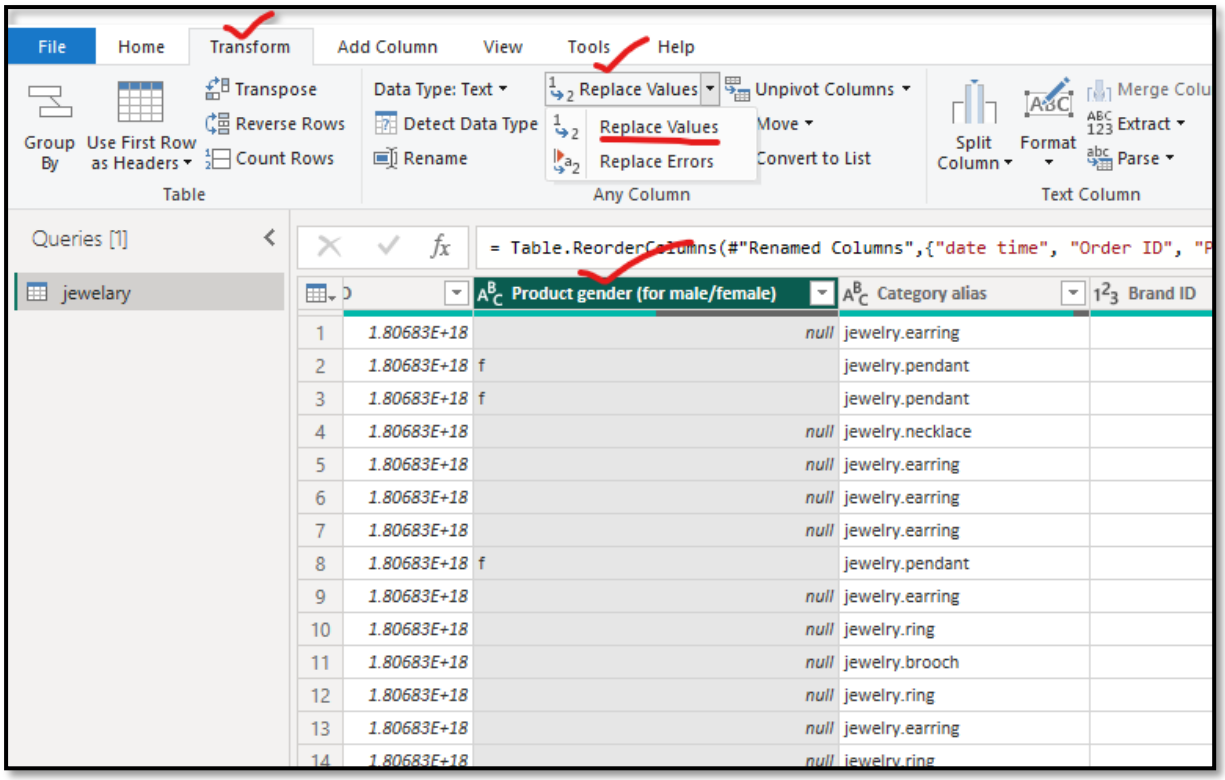


Figure-27: Replacing Null Columns

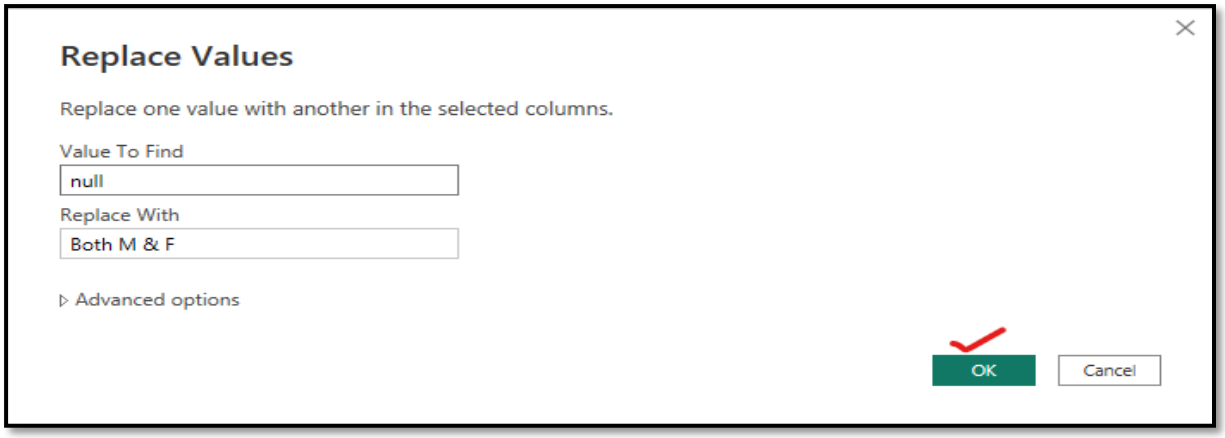


Figure-28: Replacing Null values pop up

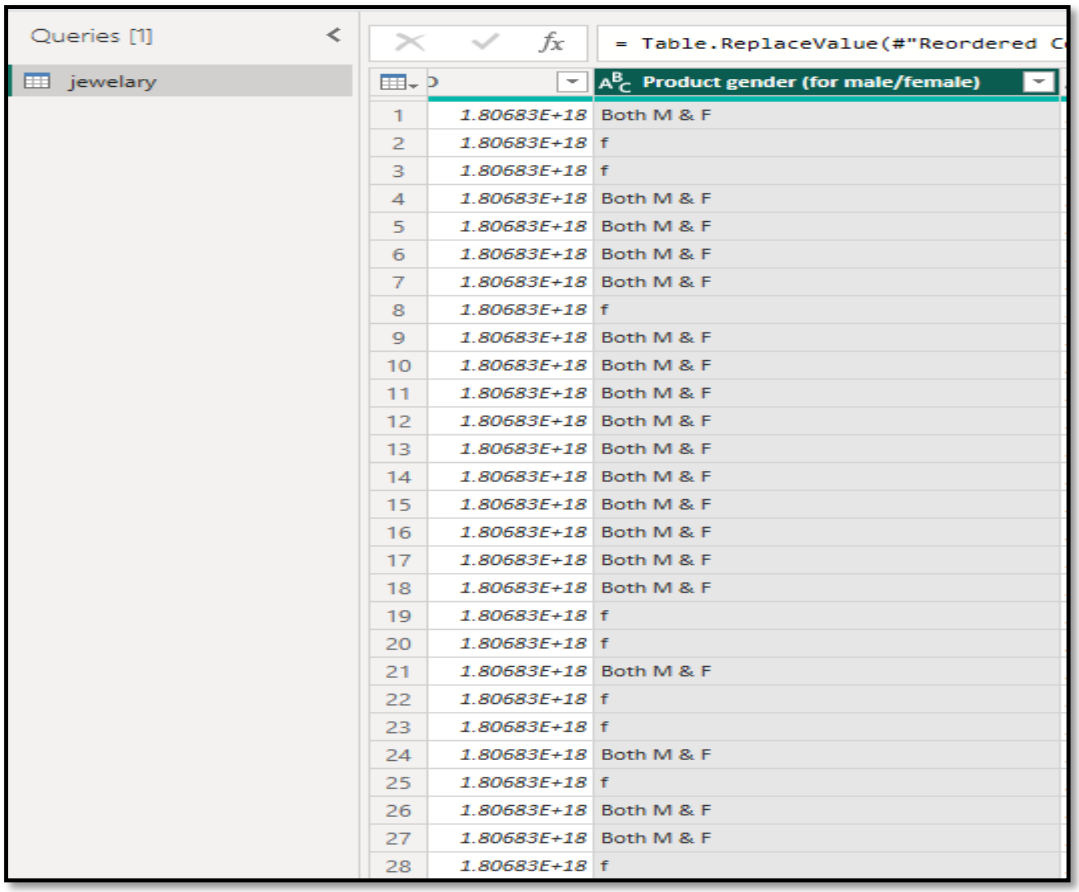
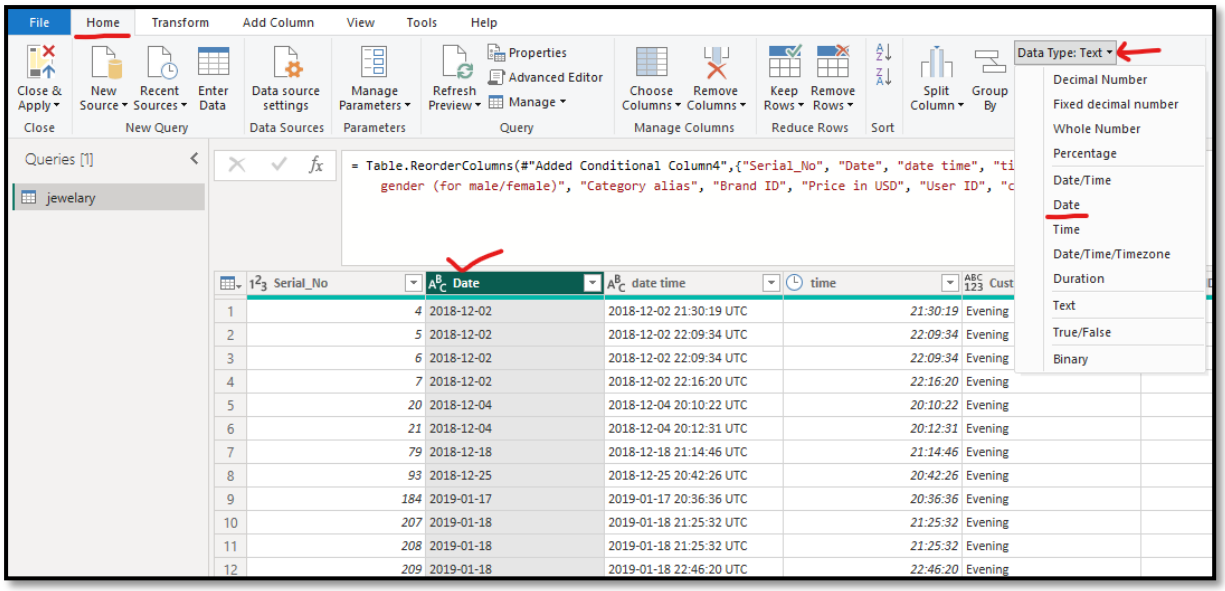


Figure-29: Replacing Null values completely

= Table.ReplaceValue(#"Reordered Columns",null,"Both M & F",Replacer.ReplaceValue,{"Product gender (for male/female)"})

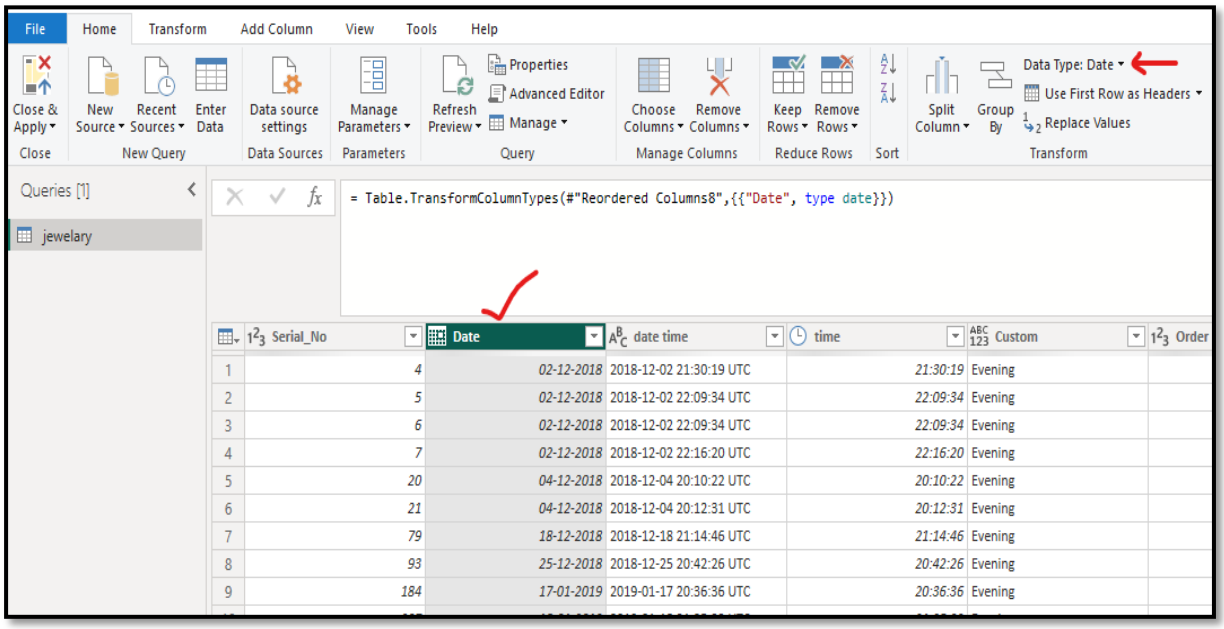
Changing Data Types of Columns

- Transaction_Date- Text → Date
- CustomerID Text → Whole number



	Serial_No	Date	date time	time	Cust
1	4	2018-12-02	2018-12-02 21:30:19 UTC	21:30:19	Evening
2	5	2018-12-02	2018-12-02 22:09:34 UTC	22:09:34	Evening
3	6	2018-12-02	2018-12-02 22:09:34 UTC	22:09:34	Evening
4	7	2018-12-02	2018-12-02 22:16:20 UTC	22:16:20	Evening
5	20	2018-12-04	2018-12-04 20:10:22 UTC	20:10:22	Evening
6	21	2018-12-04	2018-12-04 20:12:31 UTC	20:12:31	Evening
7	79	2018-12-18	2018-12-18 21:14:46 UTC	21:14:46	Evening
8	93	2018-12-25	2018-12-25 20:42:26 UTC	20:42:26	Evening
9	184	2019-01-17	2019-01-17 20:36:36 UTC	20:36:36	Evening
10	207	2019-01-18	2019-01-18 21:25:32 UTC	21:25:32	Evening
11	208	2019-01-18	2019-01-18 21:25:32 UTC	21:25:32	Evening
12	209	2019-01-18	2019-01-18 22:46:20 UTC	22:46:20	Evening

Figure-30: Data Type of 'Date'



	Serial_No	Date	date time	time	Custom
1	4	02-12-2018	2018-12-02 21:30:19 UTC	21:30:19	Evening
2	5	02-12-2018	2018-12-02 22:09:34 UTC	22:09:34	Evening
3	6	02-12-2018	2018-12-02 22:09:34 UTC	22:09:34	Evening
4	7	02-12-2018	2018-12-02 22:16:20 UTC	22:16:20	Evening
5	20	04-12-2018	2018-12-04 20:10:22 UTC	20:10:22	Evening
6	21	04-12-2018	2018-12-04 20:12:31 UTC	20:12:31	Evening
7	79	18-12-2018	2018-12-18 21:14:46 UTC	21:14:46	Evening
8	93	25-12-2018	2018-12-25 20:42:26 UTC	20:42:26	Evening
9	184	17-01-2019	2019-01-17 20:36:36 UTC	20:36:36	Evening

Figure-31: Changed Data Type of 'Date'

= Table.TransformColumnTypes(#"Reordered Columns8",{{"Date", type date}})

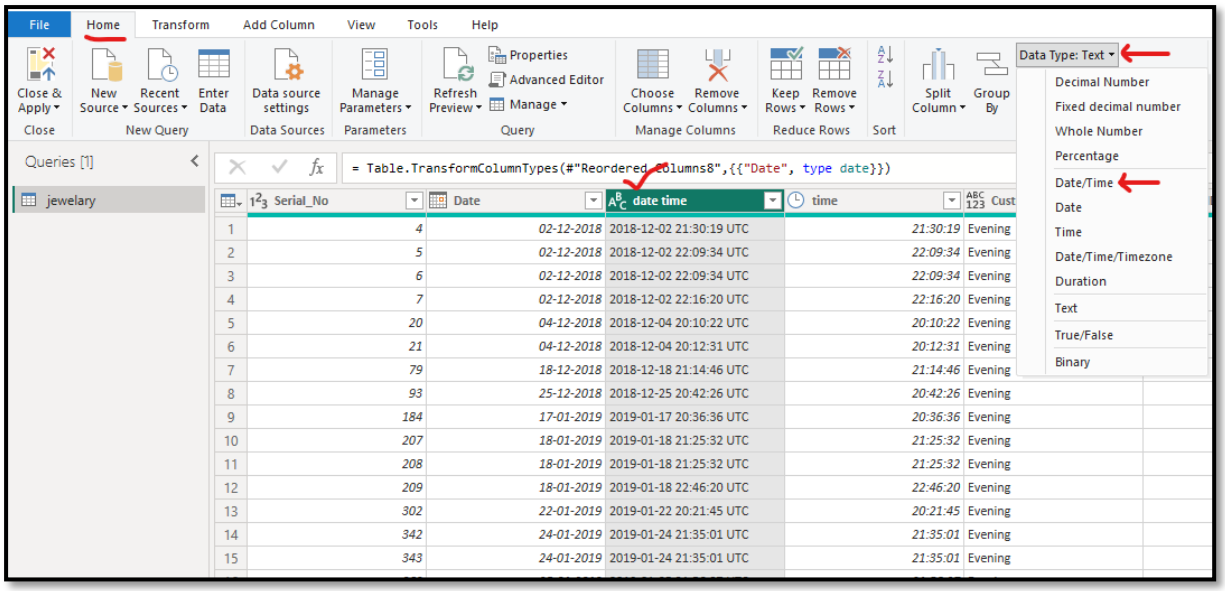


Figure-32: Data Type of 'datetime'

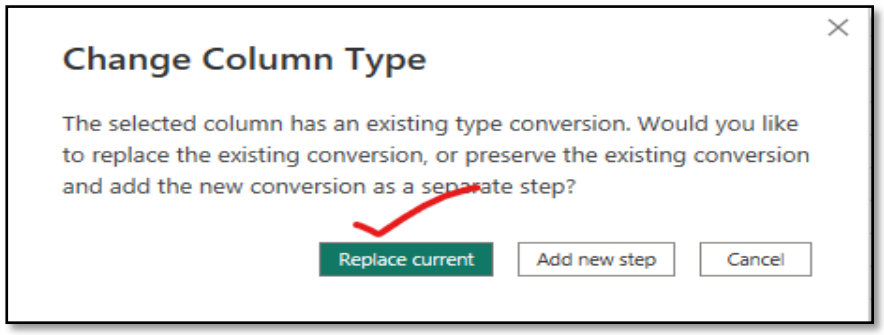


Figure-33: Data Type Changing Pop-up'

= Table.TransformColumnTypes(#"Renamed Columns2",{{"Transaction_Date", type date}, {"Date", type date}, {"CustomerID", Int64.Type}})

Adding an Index Column

An index column in Power BI DAX is a calculated column that generates an ordered sequence of integers, starting from 1. The index column acts as a unique identifier for each row in the dataset and helps perform different operations like sorting and filtering.

The index column can be added to any data model table and calculated based on the row context.

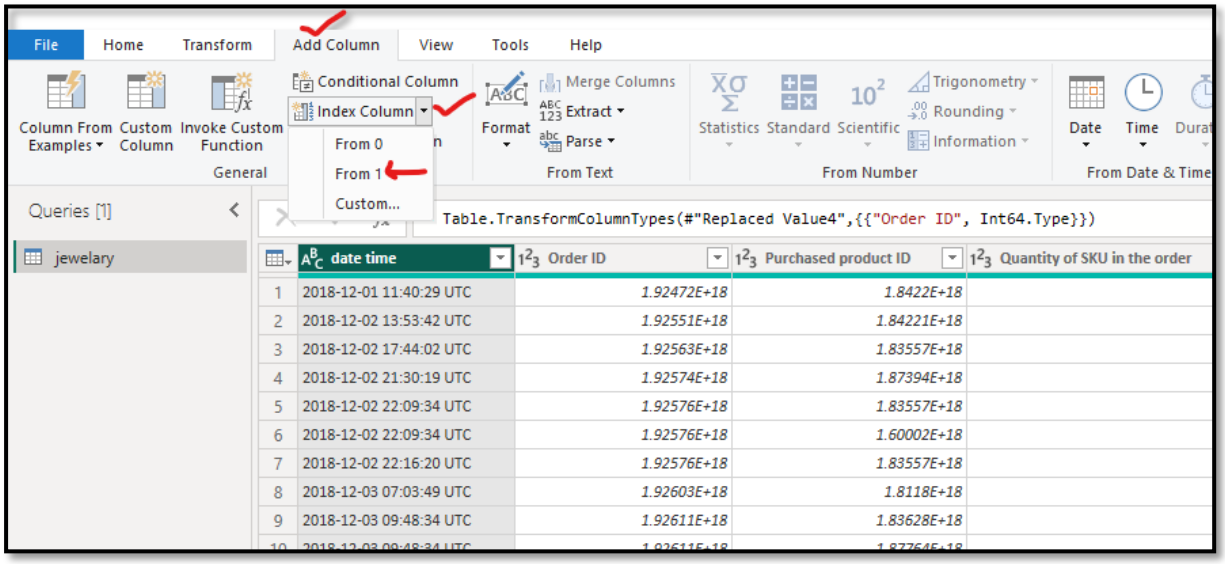


Figure-34: Add column: Index Column Option

= Table.AddIndexColumn(#"Removed Columns", "Index", 1, 1, Int64.Type)

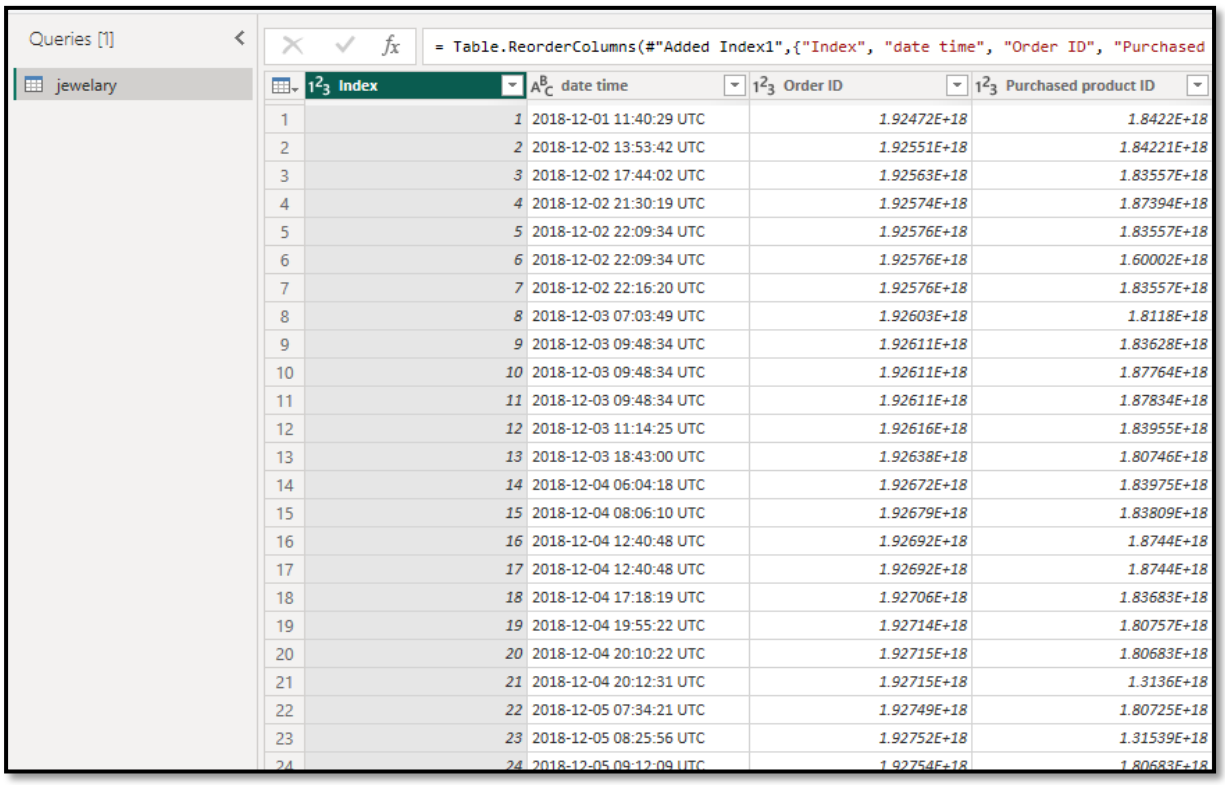
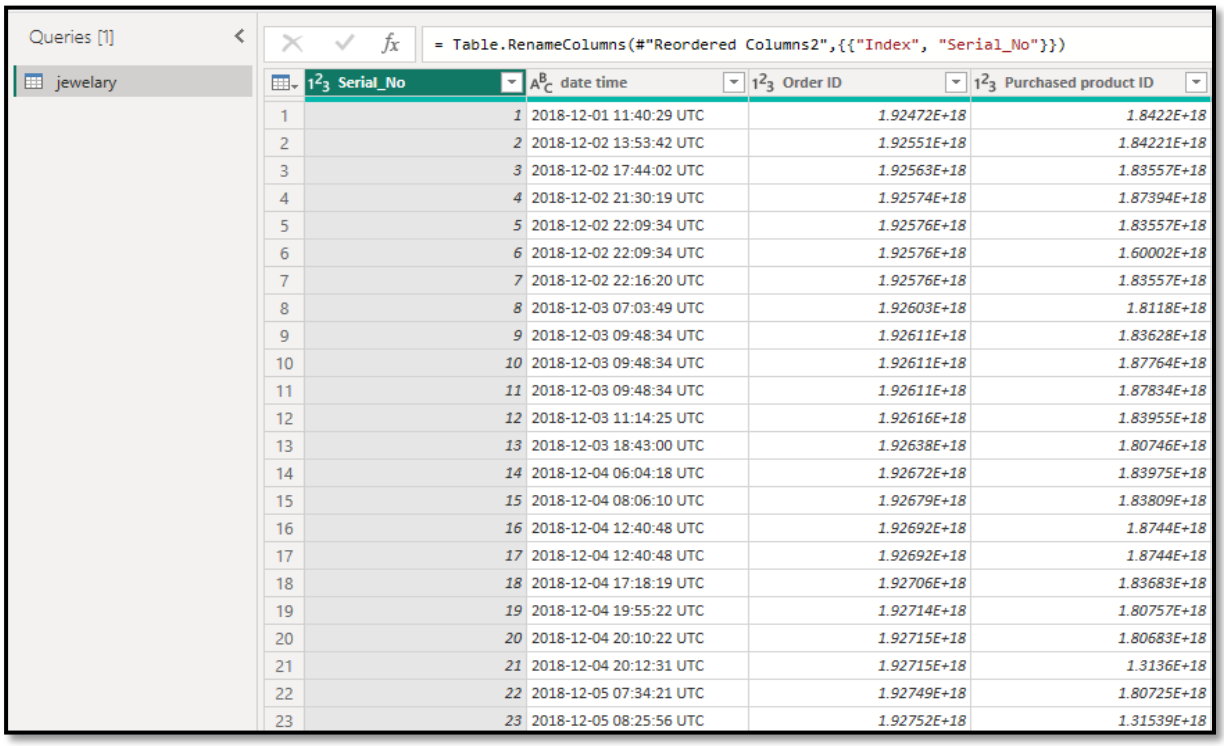


Figure-35: Index Column Added

= Table.RenameColumns(#"Reordered Columns2",{{"Index", "Serial_No"}})



Queries [1]

jewelary

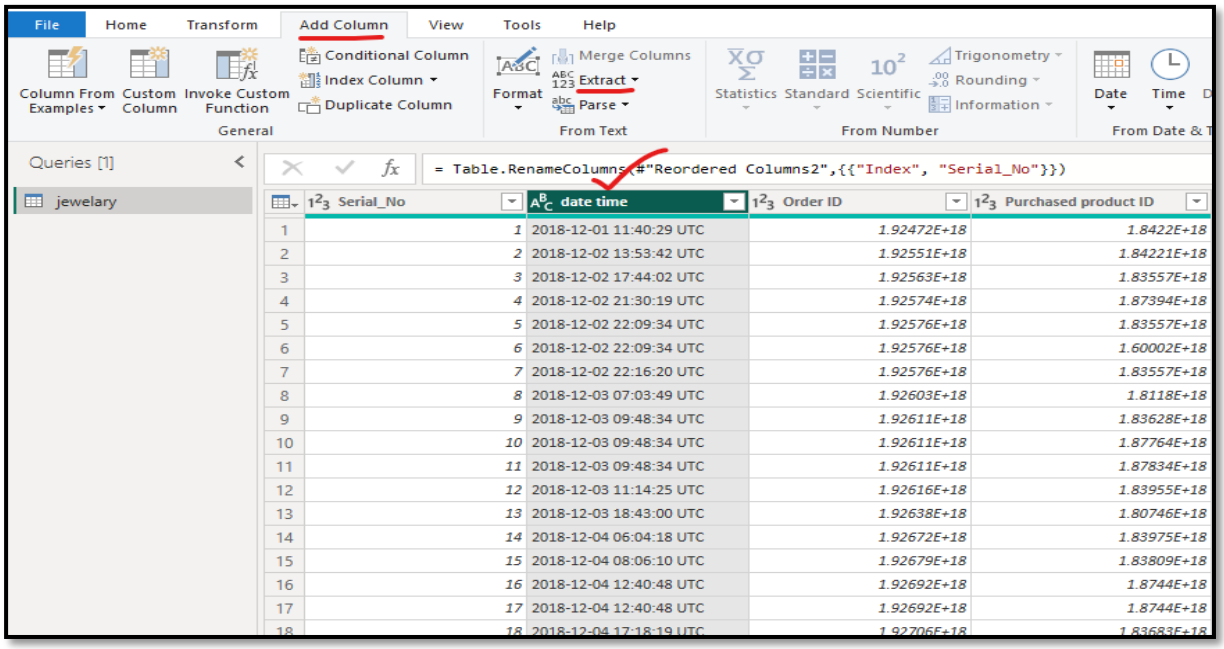
= Table.RenameColumns(#"Reordered Columns2",{{"Index", "Serial_No"}})

	Serial_No	date time	Order ID	Purchased product ID
1	1	2018-12-01 11:40:29 UTC	1.92472E+18	1.8422E+18
2	2	2018-12-02 13:53:42 UTC	1.92551E+18	1.84221E+18
3	3	2018-12-02 17:44:02 UTC	1.92563E+18	1.83557E+18
4	4	2018-12-02 21:30:19 UTC	1.92574E+18	1.87394E+18
5	5	2018-12-02 22:09:34 UTC	1.92576E+18	1.83557E+18
6	6	2018-12-02 22:09:34 UTC	1.92576E+18	1.60002E+18
7	7	2018-12-02 22:16:20 UTC	1.92576E+18	1.83557E+18
8	8	2018-12-03 07:03:49 UTC	1.92603E+18	1.8118E+18
9	9	2018-12-03 09:48:34 UTC	1.92611E+18	1.83628E+18
10	10	2018-12-03 09:48:34 UTC	1.92611E+18	1.87764E+18
11	11	2018-12-03 09:48:34 UTC	1.92611E+18	1.87834E+18
12	12	2018-12-03 11:14:25 UTC	1.92616E+18	1.83955E+18
13	13	2018-12-03 18:43:00 UTC	1.92638E+18	1.80746E+18
14	14	2018-12-04 06:04:18 UTC	1.92672E+18	1.83975E+18
15	15	2018-12-04 08:06:10 UTC	1.92679E+18	1.83809E+18
16	16	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18
17	17	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18
18	18	2018-12-04 17:18:19 UTC	1.92706E+18	1.83683E+18
19	19	2018-12-04 19:55:22 UTC	1.92714E+18	1.80757E+18
20	20	2018-12-04 20:10:22 UTC	1.92715E+18	1.80683E+18
21	21	2018-12-04 20:12:31 UTC	1.92715E+18	1.3136E+18
22	22	2018-12-05 07:34:21 UTC	1.92749E+18	1.80725E+18
23	23	2018-12-05 08:25:56 UTC	1.92752E+18	1.31539E+18

Figure-36: Name change of Index Column

Extracting Column

The ‘datetime’ column has date and time both values. In order to separate the date and time, we simply go to ‘Add Column’ then click on ‘Extract’.



File Home Transform Add Column View Tools Help

Column From Examples | Custom Column | Invoke Custom Function | Conditional Column | Index Column | Duplicate Column — General

Format | Merge Columns | Extract | Parse — From Text

Statistics | Standard | Scientific | Trigonometry | Rounding | Information — From Number

Date | Time — From Date & T

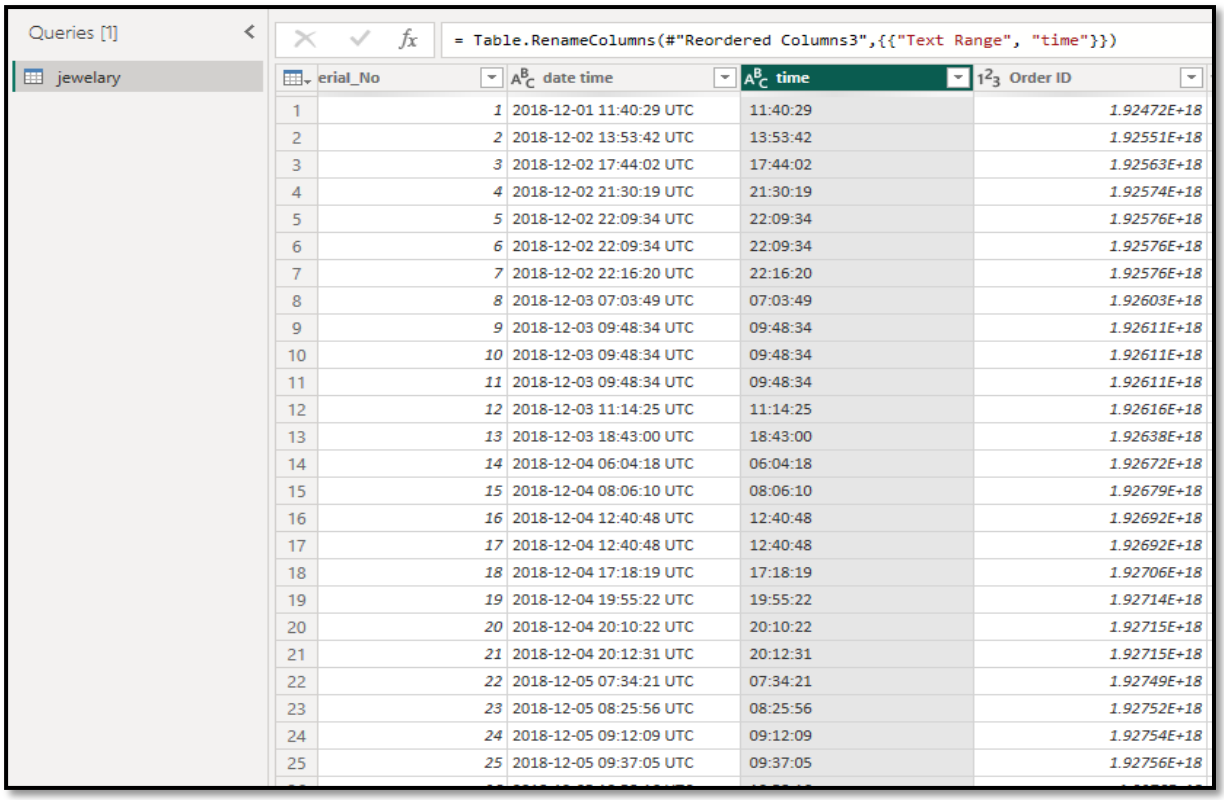
Queries [1]

jewelary

= Table.RenameColumns(#"Reordered Columns2",{{"Index", "Serial_No"}})

	Serial_No	date time	Order ID	Purchased product ID
1	1	2018-12-01 11:40:29 UTC	1.92472E+18	1.8422E+18
2	2	2018-12-02 13:53:42 UTC	1.92551E+18	1.84221E+18
3	3	2018-12-02 17:44:02 UTC	1.92563E+18	1.83557E+18
4	4	2018-12-02 21:30:19 UTC	1.92574E+18	1.87394E+18
5	5	2018-12-02 22:09:34 UTC	1.92576E+18	1.83557E+18
6	6	2018-12-02 22:09:34 UTC	1.92576E+18	1.60002E+18
7	7	2018-12-02 22:16:20 UTC	1.92576E+18	1.83557E+18
8	8	2018-12-03 07:03:49 UTC	1.92603E+18	1.8118E+18
9	9	2018-12-03 09:48:34 UTC	1.92611E+18	1.83628E+18
10	10	2018-12-03 09:48:34 UTC	1.92611E+18	1.87764E+18
11	11	2018-12-03 09:48:34 UTC	1.92611E+18	1.87834E+18
12	12	2018-12-03 11:14:25 UTC	1.92616E+18	1.83955E+18
13	13	2018-12-03 18:43:00 UTC	1.92638E+18	1.80746E+18
14	14	2018-12-04 06:04:18 UTC	1.92672E+18	1.83975E+18
15	15	2018-12-04 08:06:10 UTC	1.92679E+18	1.83809E+18
16	16	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18
17	17	2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18
18	18	2018-12-04 17:18:19 UTC	1.92706E+18	1.83683E+18

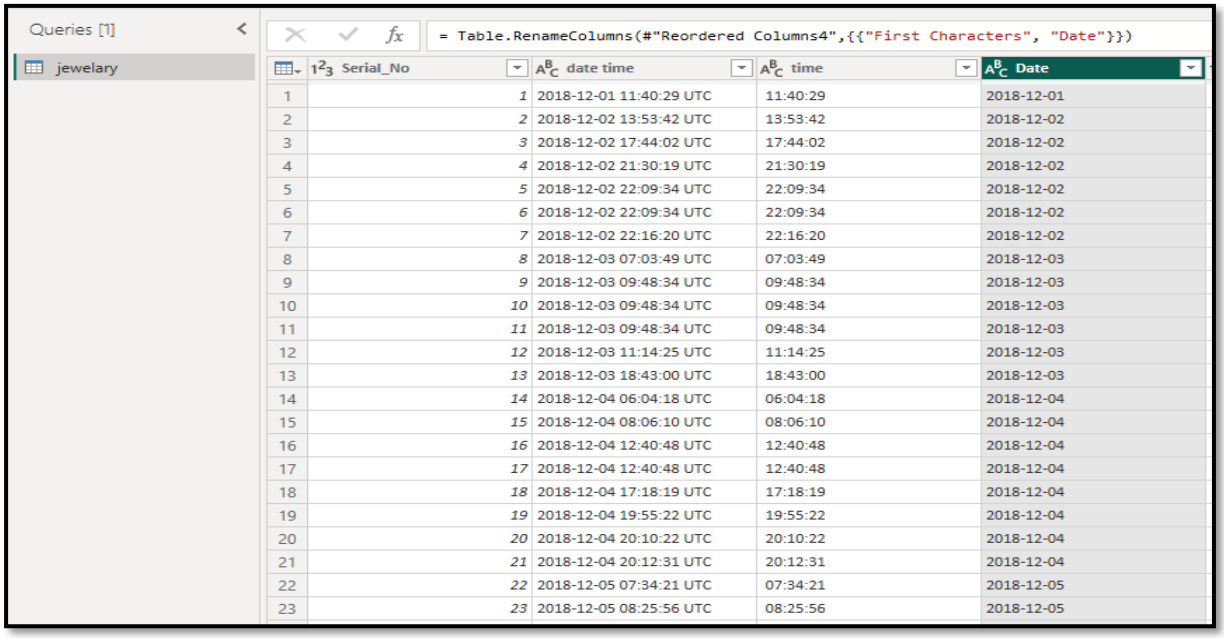
Figure-37. Extracting Column option



	erial_No	date time	time	Order ID
1	1	2018-12-01 11:40:29 UTC	11:40:29	1.92472E+18
2	2	2018-12-02 13:53:42 UTC	13:53:42	1.92551E+18
3	3	2018-12-02 17:44:02 UTC	17:44:02	1.92563E+18
4	4	2018-12-02 21:30:19 UTC	21:30:19	1.92574E+18
5	5	2018-12-02 22:09:34 UTC	22:09:34	1.92576E+18
6	6	2018-12-02 22:09:34 UTC	22:09:34	1.92576E+18
7	7	2018-12-02 22:16:20 UTC	22:16:20	1.92576E+18
8	8	2018-12-03 07:03:49 UTC	07:03:49	1.92603E+18
9	9	2018-12-03 09:48:34 UTC	09:48:34	1.92611E+18
10	10	2018-12-03 09:48:34 UTC	09:48:34	1.92611E+18
11	11	2018-12-03 09:48:34 UTC	09:48:34	1.92611E+18
12	12	2018-12-03 11:14:25 UTC	11:14:25	1.92616E+18
13	13	2018-12-03 18:43:00 UTC	18:43:00	1.92638E+18
14	14	2018-12-04 06:04:18 UTC	06:04:18	1.92672E+18
15	15	2018-12-04 08:06:10 UTC	08:06:10	1.92679E+18
16	16	2018-12-04 12:40:48 UTC	12:40:48	1.92692E+18
17	17	2018-12-04 12:40:48 UTC	12:40:48	1.92692E+18
18	18	2018-12-04 17:18:19 UTC	17:18:19	1.92706E+18
19	19	2018-12-04 19:55:22 UTC	19:55:22	1.92714E+18
20	20	2018-12-04 20:10:22 UTC	20:10:22	1.92715E+18
21	21	2018-12-04 20:12:31 UTC	20:12:31	1.92715E+18
22	22	2018-12-05 07:34:21 UTC	07:34:21	1.92749E+18
23	23	2018-12-05 08:25:56 UTC	08:25:56	1.92752E+18
24	24	2018-12-05 09:12:09 UTC	09:12:09	1.92754E+18
25	25	2018-12-05 09:37:05 UTC	09:37:05	1.92756E+18

Figure-38. Extracting Column: time

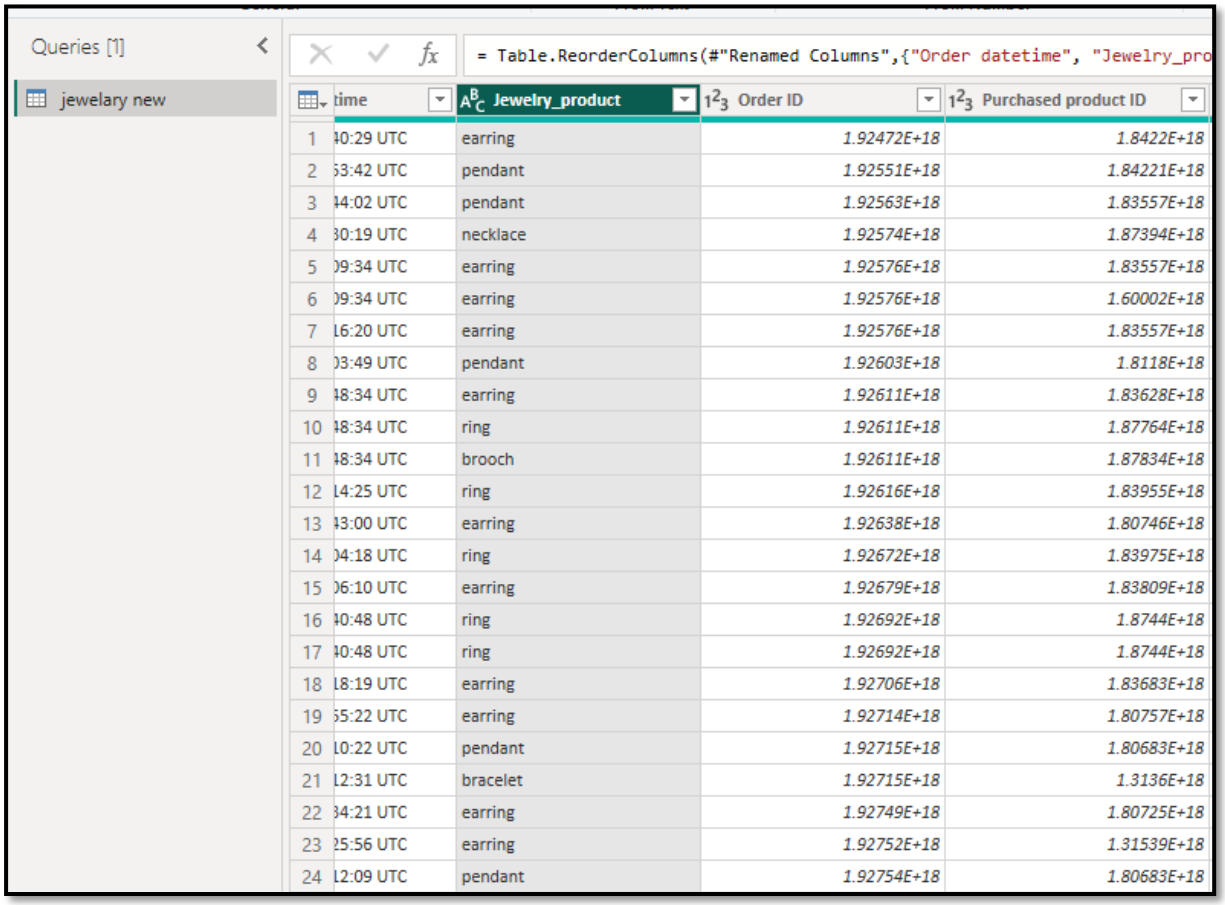
= Table.AddColumn(#"Removed Columns1", "Text Range", each Text.Middle([date time], 10, 10), type text)



	Serial_No	date time	time	Date
1	1	2018-12-01 11:40:29 UTC	11:40:29	2018-12-01
2	2	2018-12-02 13:53:42 UTC	13:53:42	2018-12-02
3	3	2018-12-02 17:44:02 UTC	17:44:02	2018-12-02
4	4	2018-12-02 21:30:19 UTC	21:30:19	2018-12-02
5	5	2018-12-02 22:09:34 UTC	22:09:34	2018-12-02
6	6	2018-12-02 22:09:34 UTC	22:09:34	2018-12-02
7	7	2018-12-02 22:16:20 UTC	22:16:20	2018-12-02
8	8	2018-12-03 07:03:49 UTC	07:03:49	2018-12-03
9	9	2018-12-03 09:48:34 UTC	09:48:34	2018-12-03
10	10	2018-12-03 09:48:34 UTC	09:48:34	2018-12-03
11	11	2018-12-03 09:48:34 UTC	09:48:34	2018-12-03
12	12	2018-12-03 11:14:25 UTC	11:14:25	2018-12-03
13	13	2018-12-03 18:43:00 UTC	18:43:00	2018-12-03
14	14	2018-12-04 06:04:18 UTC	06:04:18	2018-12-04
15	15	2018-12-04 08:06:10 UTC	08:06:10	2018-12-04
16	16	2018-12-04 12:40:48 UTC	12:40:48	2018-12-04
17	17	2018-12-04 12:40:48 UTC	12:40:48	2018-12-04
18	18	2018-12-04 17:18:19 UTC	17:18:19	2018-12-04
19	19	2018-12-04 19:55:22 UTC	19:55:22	2018-12-04
20	20	2018-12-04 20:10:22 UTC	20:10:22	2018-12-04
21	21	2018-12-04 20:12:31 UTC	20:12:31	2018-12-04
22	22	2018-12-05 07:34:21 UTC	07:34:21	2018-12-05
23	23	2018-12-05 08:25:56 UTC	08:25:56	2018-12-05

Figure-39. Extracting Column: date

= Table.AddColumn(#"Renamed Columns2", "First Characters", each Text.Start([date time], 10), type text)



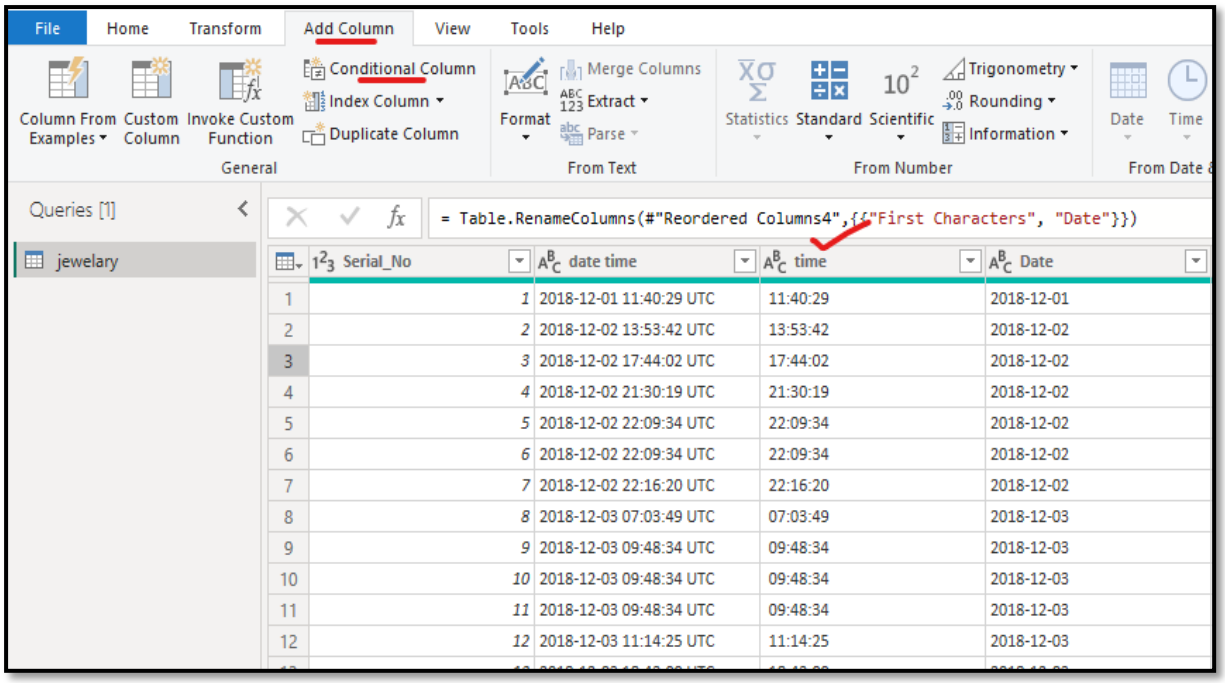
	time	Jewelry_product	Order ID	Purchased product ID
1	40:29 UTC	earring	1.92472E+18	1.8422E+18
2	53:42 UTC	pendant	1.92551E+18	1.84221E+18
3	44:02 UTC	pendant	1.92563E+18	1.83557E+18
4	30:19 UTC	necklace	1.92574E+18	1.87394E+18
5	09:34 UTC	earring	1.92576E+18	1.83557E+18
6	09:34 UTC	earring	1.92576E+18	1.60002E+18
7	16:20 UTC	earring	1.92576E+18	1.83557E+18
8	03:49 UTC	pendant	1.92603E+18	1.8118E+18
9	48:34 UTC	earring	1.92611E+18	1.83628E+18
10	48:34 UTC	ring	1.92611E+18	1.87764E+18
11	48:34 UTC	brooch	1.92611E+18	1.87834E+18
12	14:25 UTC	ring	1.92616E+18	1.83955E+18
13	43:00 UTC	earring	1.92638E+18	1.80746E+18
14	04:18 UTC	ring	1.92672E+18	1.83975E+18
15	06:10 UTC	earring	1.92679E+18	1.83809E+18
16	40:48 UTC	ring	1.92692E+18	1.8744E+18
17	40:48 UTC	ring	1.92692E+18	1.8744E+18
18	18:19 UTC	earring	1.92706E+18	1.83683E+18
19	55:22 UTC	earring	1.92714E+18	1.80757E+18
20	10:22 UTC	pendant	1.92715E+18	1.80683E+18
21	12:31 UTC	bracelet	1.92715E+18	1.3136E+18
22	34:21 UTC	earring	1.92749E+18	1.80725E+18
23	25:56 UTC	earring	1.92752E+18	1.31539E+18
24	12:09 UTC	pendant	1.92754E+18	1.80683E+18

Figure-40. Extracting Column: Jewelry_product

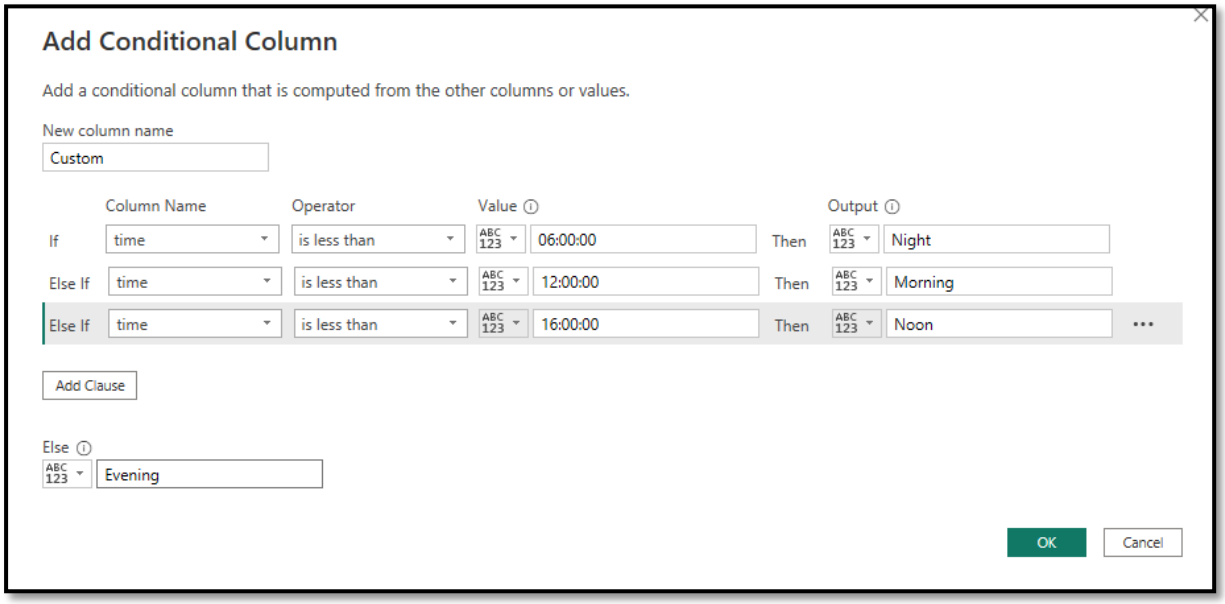
= Table.AddColumn(#"Replaced Value4", "Text After Delimiter", each Text.AfterDelimiter([Column6], "."), type text)

Conditional Column

By conditional column, we can manage the time of order to the duration of the day like, night, noon, morning and evening.



Figue-41. Conditional column option



Figue-42. Conditional column values

= Table.AddColumn(#"Removed Columns6", "Custom", each if [time] > #time(6, 0, 0) then "Morning" else if [time] > #time(12, 0, 0) then "Noon" else if [time] > #time(16, 0, 0) then "Evening" else "Night")

Bi Data Modeling

Fact and Dimension in Datasheet

Dimension Table:

- descriptive information about the data/information. Time, location, product, customer, and employee are a few examples of dimensions.
- The dimension table is a crucial part of the Star Schema. The dimensions along which the values of the attributes are taken in the fact table are included in a dimension table.

Fact Table:

- Quantitative data or metrics, also known as facts. These facts are typically numerical measures, such as sales revenue, quantity sold, or profit.
- These relationships enable analysts to combine quantitative data with descriptive attributes for comprehensive analysis.

Fact and Dimension Table

Dimension Dim_Order_Info

The “Dim_Order_Info” table contains the following columns -

a) Serial_No
b) Order ID
c) Order datetime
d) order_Time
e) Order_Date
f) Day_Duration

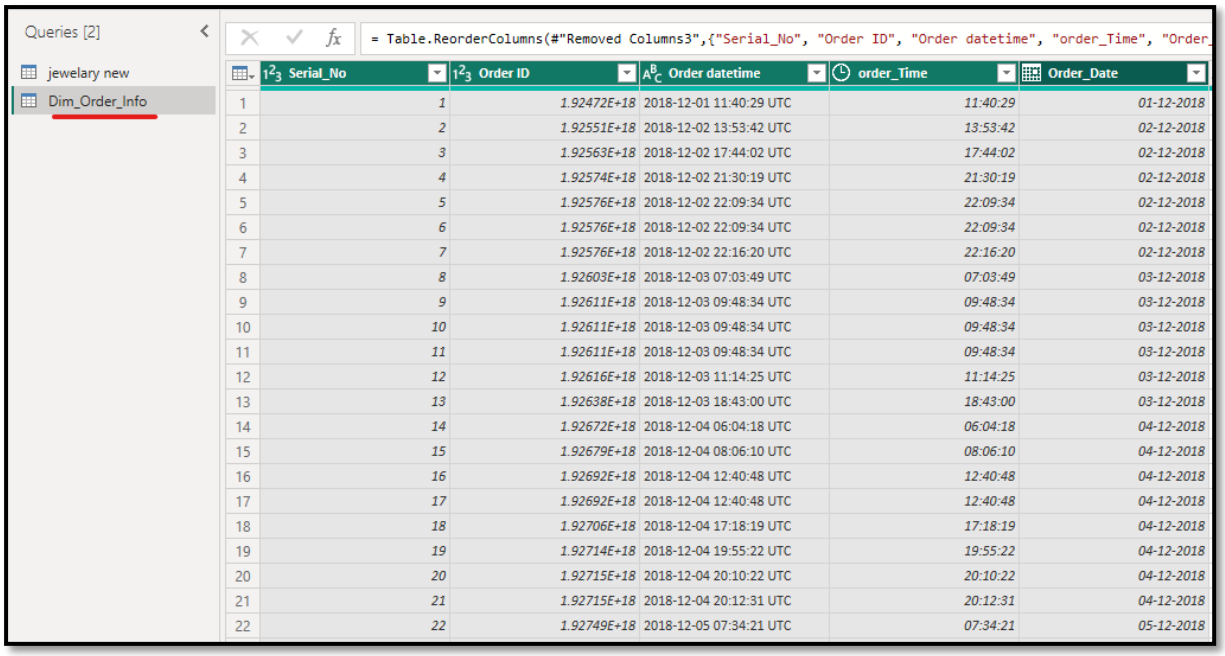


Figure-43: Creation of Dimension Dim_Order_Info Table

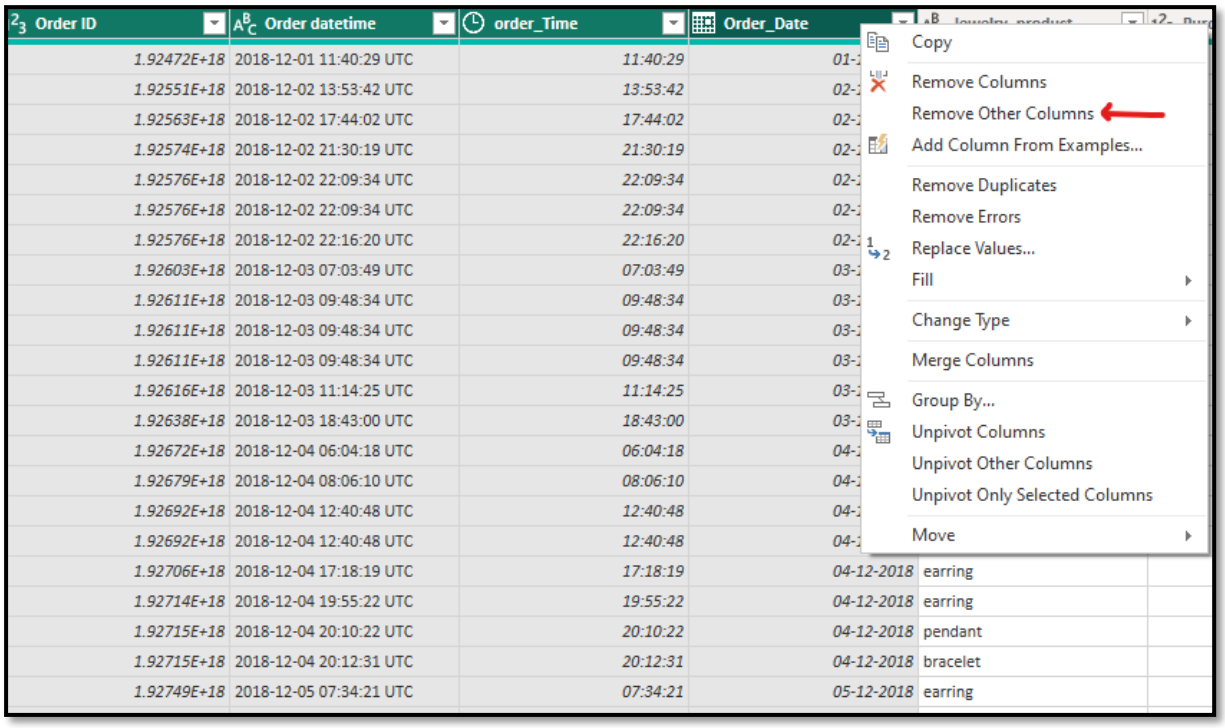
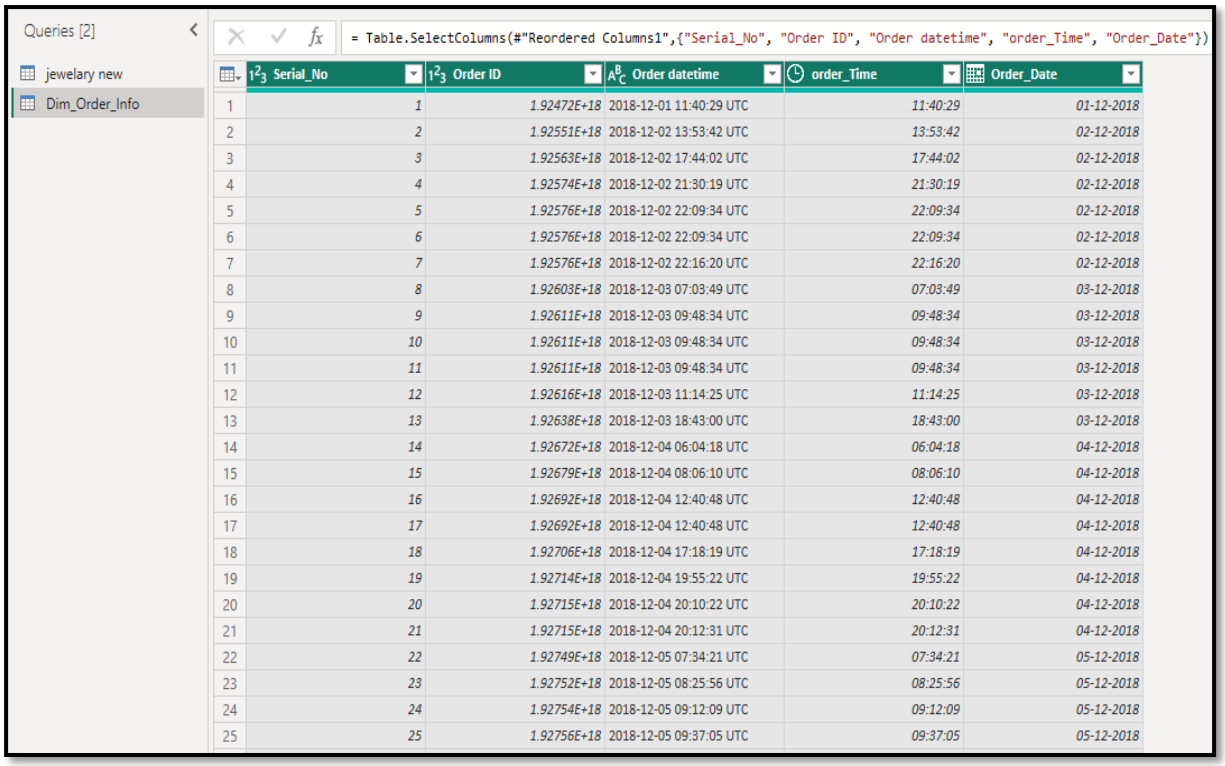


Figure-44: Dimension Dim_Order_Info (Removing extra column)



	Serial_No	Order ID	Order datetime	order_Time	Order_Date
1	1	1.92472E+18	2018-12-01 11:40:29 UTC	11:40:29	01-12-2018
2	2	1.92551E+18	2018-12-02 13:53:42 UTC	13:53:42	02-12-2018
3	3	1.92563E+18	2018-12-02 17:44:02 UTC	17:44:02	02-12-2018
4	4	1.92574E+18	2018-12-02 21:30:19 UTC	21:30:19	02-12-2018
5	5	1.92576E+18	2018-12-02 22:09:34 UTC	22:09:34	02-12-2018
6	6	1.92576E+18	2018-12-02 22:09:34 UTC	22:09:34	02-12-2018
7	7	1.92576E+18	2018-12-02 22:16:20 UTC	22:16:20	02-12-2018
8	8	1.92603E+18	2018-12-03 07:03:49 UTC	07:03:49	03-12-2018
9	9	1.92611E+18	2018-12-03 09:48:34 UTC	09:48:34	03-12-2018
10	10	1.92611E+18	2018-12-03 09:48:34 UTC	09:48:34	03-12-2018
11	11	1.92611E+18	2018-12-03 09:48:34 UTC	09:48:34	03-12-2018
12	12	1.92616E+18	2018-12-03 11:14:25 UTC	11:14:25	03-12-2018
13	13	1.92638E+18	2018-12-03 18:43:00 UTC	18:43:00	03-12-2018
14	14	1.92672E+18	2018-12-04 06:04:18 UTC	06:04:18	04-12-2018
15	15	1.92679E+18	2018-12-04 08:06:10 UTC	08:06:10	04-12-2018
16	16	1.92692E+18	2018-12-04 12:40:48 UTC	12:40:48	04-12-2018
17	17	1.92692E+18	2018-12-04 12:40:48 UTC	12:40:48	04-12-2018
18	18	1.92706E+18	2018-12-04 17:18:19 UTC	17:18:19	04-12-2018
19	19	1.92714E+18	2018-12-04 19:55:22 UTC	19:55:22	04-12-2018
20	20	1.92715E+18	2018-12-04 20:10:22 UTC	20:10:22	04-12-2018
21	21	1.92715E+18	2018-12-04 20:12:31 UTC	20:12:31	04-12-2018
22	22	1.92749E+18	2018-12-05 07:34:21 UTC	07:34:21	05-12-2018
23	23	1.92752E+18	2018-12-05 08:25:56 UTC	08:25:56	05-12-2018
24	24	1.92754E+18	2018-12-05 09:12:09 UTC	09:12:09	05-12-2018
25	25	1.92756E+18	2018-12-05 09:37:05 UTC	09:37:05	05-12-2018

Figure-45: Dimension Dim_Order_Info

= Table.SelectColumns(#"Reordered Columns1",{"Serial_No", "Order ID", "Order datetime", "order_Time", "Order_Date"})

Dimension Dim_Prod_Info

The “Dim_Prod_Info” table contains the following columns-

a) Serial_No
b) Color
c) Jewelry_product
d) Product gender (for male/female)
e) Category alias
f) Metal
g) Stone_used

	Serial_No	Jewelry_product	Product gender (for male/female)	Color	Stone_used	m
1	1	earring	Both M & F	red	diamond	gold
2	2	pendant	F	white	sapphire	gold
3	3	pendant	F	red	diamond	gold
4	4	necklace	Both M & F	red	amethyst	gold
5	5	earring	Both M & F	red	No Stone	gold
6	6	earring	Both M & F	red	diamond	gold
7	7	earring	Both M & F	white	fianit	gold
8	8	pendant	F	red	pearl	gold
9	9	earring	Both M & F	red	diamond	gold
10	10	ring	Both M & F	white	diamond	gold
11	11	brooch	Both M & F	red	fianit	gold
12	12	ring	Both M & F	yellow	No Stone	gold
13	13	earring	Both M & F	white	fianit	gold

Figure-46: Creation of Dimension Dim_Prod_Info Table

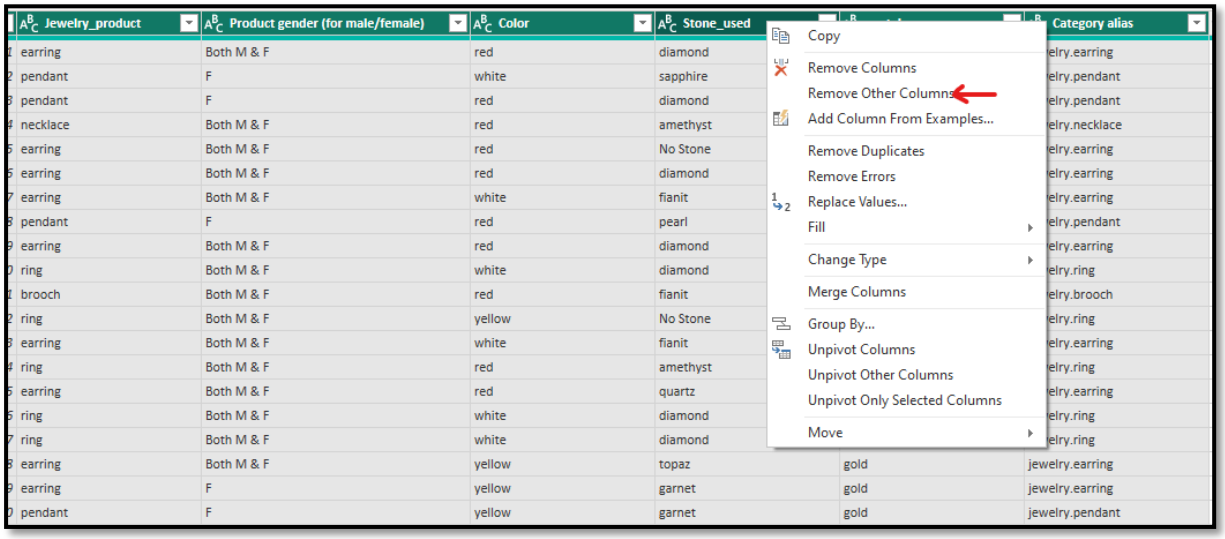
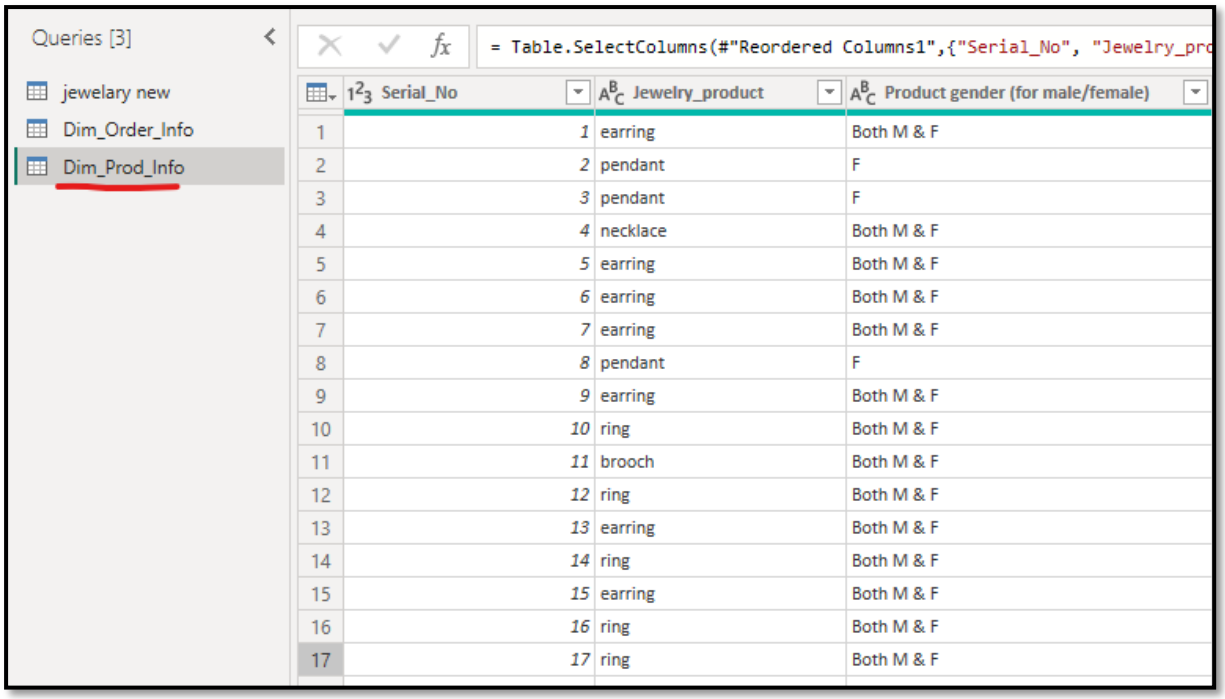


Figure-47: Dimension Dim_Prod_Info (Removing extra column)



	Serial_No	Jewelry_product	Product gender (for male/female)
1	1	earring	Both M & F
2	2	pendant	F
3	3	pendant	F
4	4	necklace	Both M & F
5	5	earring	Both M & F
6	6	earring	Both M & F
7	7	earring	Both M & F
8	8	pendant	F
9	9	earring	Both M & F
10	10	ring	Both M & F
11	11	brooch	Both M & F
12	12	ring	Both M & F
13	13	earring	Both M & F
14	14	ring	Both M & F
15	15	earring	Both M & F
16	16	ring	Both M & F
17	17	ring	Both M & F

Figure-48: Dimension Dim_Prod_Info

= Table.SelectColumns(#"Reordered Columns1",{"Serial_No", "Jewelry_product", "Product gender (for male/female)", "Color", "Stone_used", "metal", "Category alias"})

Dimension Dim_Cust_Info

The “Dim_ Cust_Info” table contains the following columns-

a) Serial_No
b) Category ID
c) Purchased product ID
d) User ID
e) Brand ID

Serial_No	Brand ID	User ID	Purchased product ID	Category ID
1	0	1.51592E+18	1.8422E+18	1.80683E+18
2	1	1.51592E+18	1.84221E+18	1.80683E+18
3	0	1.51592E+18	1.83557E+18	1.80683E+18
4	0	1.51592E+18	1.87394E+18	1.80683E+18
5	1	1.51592E+18	1.83557E+18	1.80683E+18
6	0	1.51592E+18	1.60002E+18	1.80683E+18
7	0	1.51592E+18	1.83557E+18	1.80683E+18
8	2	1.51592E+18	1.8118E+18	1.80683E+18
9	1	1.51592E+18	1.83628E+18	1.80683E+18
10	1	1.51592E+18	1.87764E+18	1.80683E+18
11	1	1.51592E+18	1.87834E+18	1.80683E+18
12	1	1.51592E+18	1.83955E+18	1.80683E+18
13	0	1.51592E+18	1.80746E+18	1.80683E+18
14	0	1.51592E+18	1.83975E+18	1.80683E+18
15	0	1.51592E+18	1.83809E+18	1.80683E+18
16	1	1.51592E+18	1.8744E+18	1.80683E+18
17	1	1.51592E+18	1.8744E+18	1.80683E+18
18	0	1.51592E+18	1.83683E+18	1.80683E+18
19	1	1.51592E+18	1.80757E+18	1.80683E+18
20	1	1.51592E+18	1.80683E+18	1.80683E+18
21	0	1.51592E+18	1.3136E+18	1.80683E+18
22	2	1.51592E+18	1.80725E+18	1.80683E+18
23	1	1.48112E+18	1.31539E+18	1.80683E+18

Figure-49: Creation of Dimension Dim_Cust Table

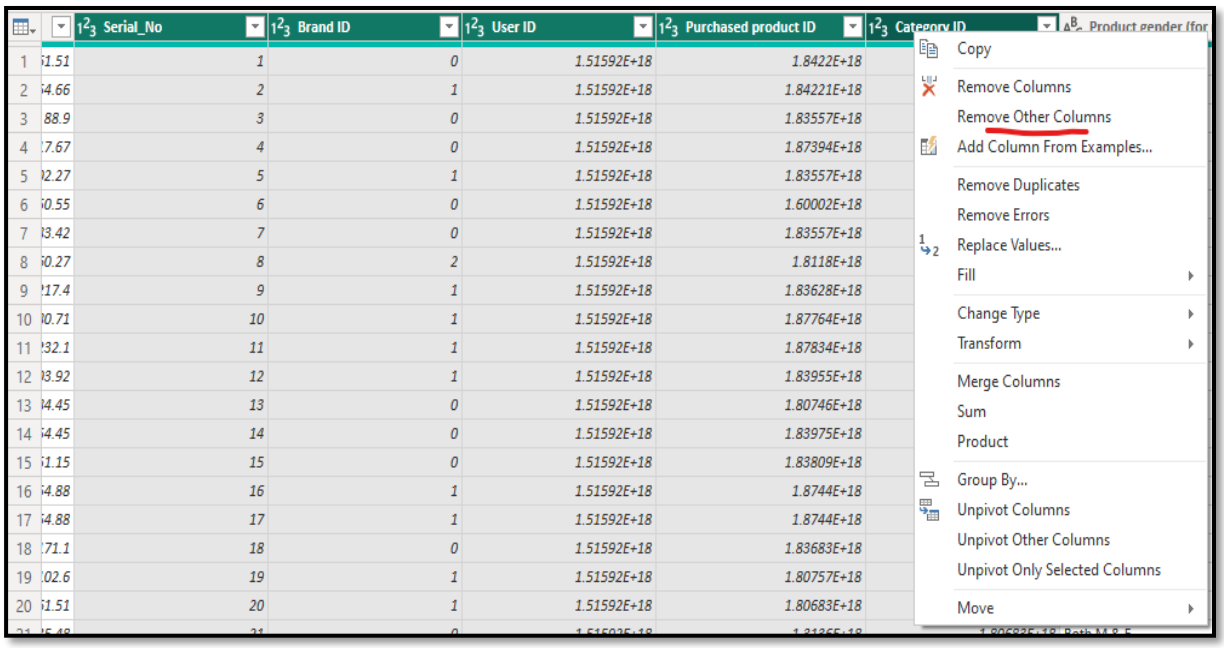


Figure-50: Dimension Dim_Cust_Info (Removing extra column)

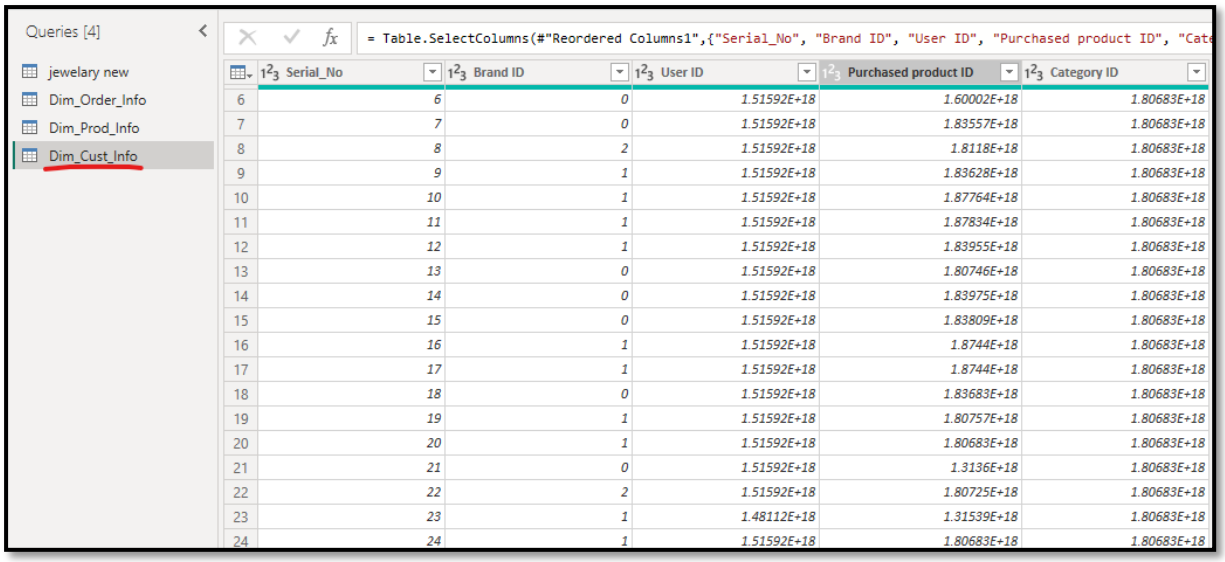


Figure-51: Dimension Dim_Cust_Info

= Table.SelectColumns(#"Reordered Columns1",{"Serial_No", "Brand ID", "User ID", "Purchased product ID", "Category ID"})

Dimension Dim_Jewelry_Info

The "Dim_Jewelry_Info" table contains the following columns-

a) Serial_No
b) Metal
c) Jewelry_product
d) Category alias

Serial_No	Category alias	Jewelry_product	metal	
1	jewelry.earring	earring	gold	B
2	jewelry.pendant	pendant	gold	F
3	jewelry.pendant	pendant	gold	F
4	jewelry.necklace	necklace	gold	B
5	jewelry.earring	earring	gold	B
6	jewelry.earring	earring	gold	B
7	jewelry.earring	earring	gold	B
8	jewelry.pendant	pendant	gold	F
9	jewelry.earring	earring	gold	B
10	jewelry.ring	ring	gold	B
11	jewelry.brooch	brooch	gold	B
12	jewelry.ring	ring	gold	B
13	jewelry.earring	earring	gold	B
14	jewelry.ring	ring	gold	B
15	jewelry.earring	earring	gold	B
16	jewelry.ring	ring	gold	B
17	jewelry.ring	ring	gold	B
18	jewelry.earring	earring	gold	B
19	jewelry.earring	earring	gold	F
20	jewelry.pendant	pendant	gold	F
21	jewelry.bracelet	bracelet	gold	B
22	jewelry.earring	earring	gold	F

Figure-52: Creation of Dimension Dim_Jewelry_Info Table

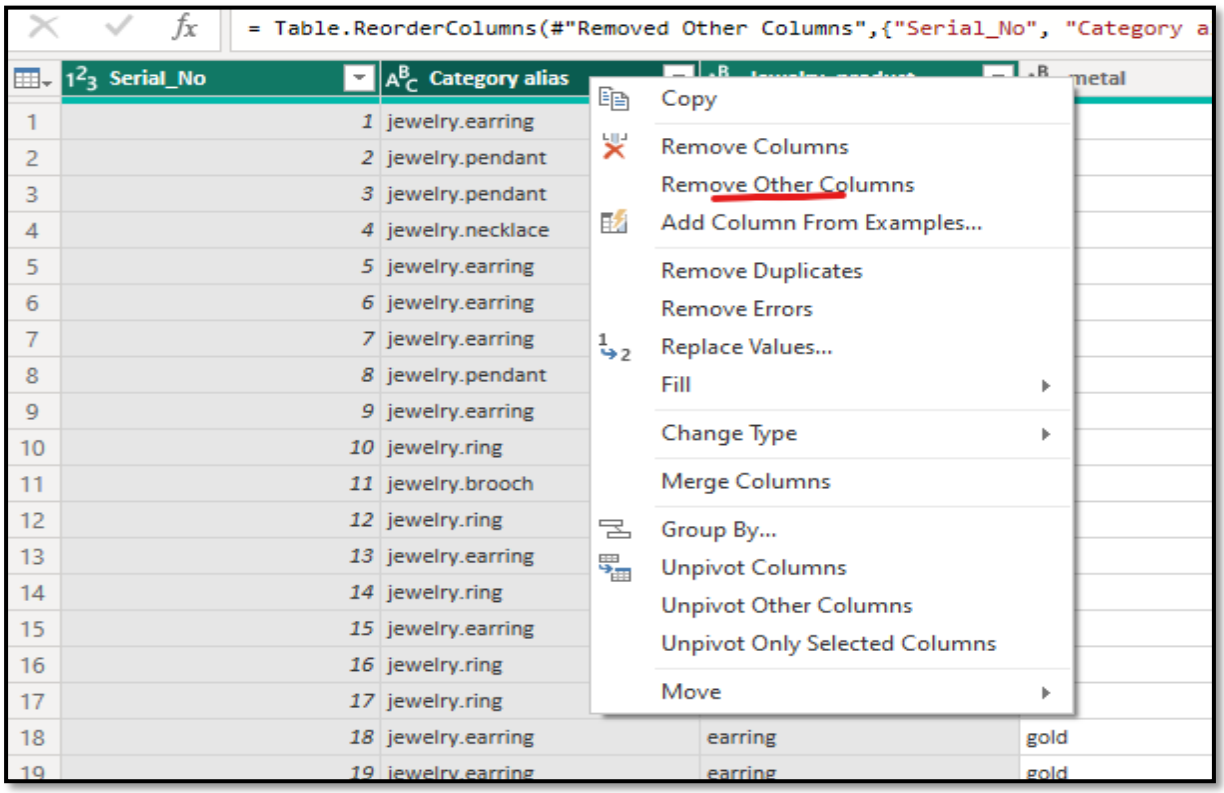


Figure-53: Dimension Dim_Jewelry_Info (Removing extra column)

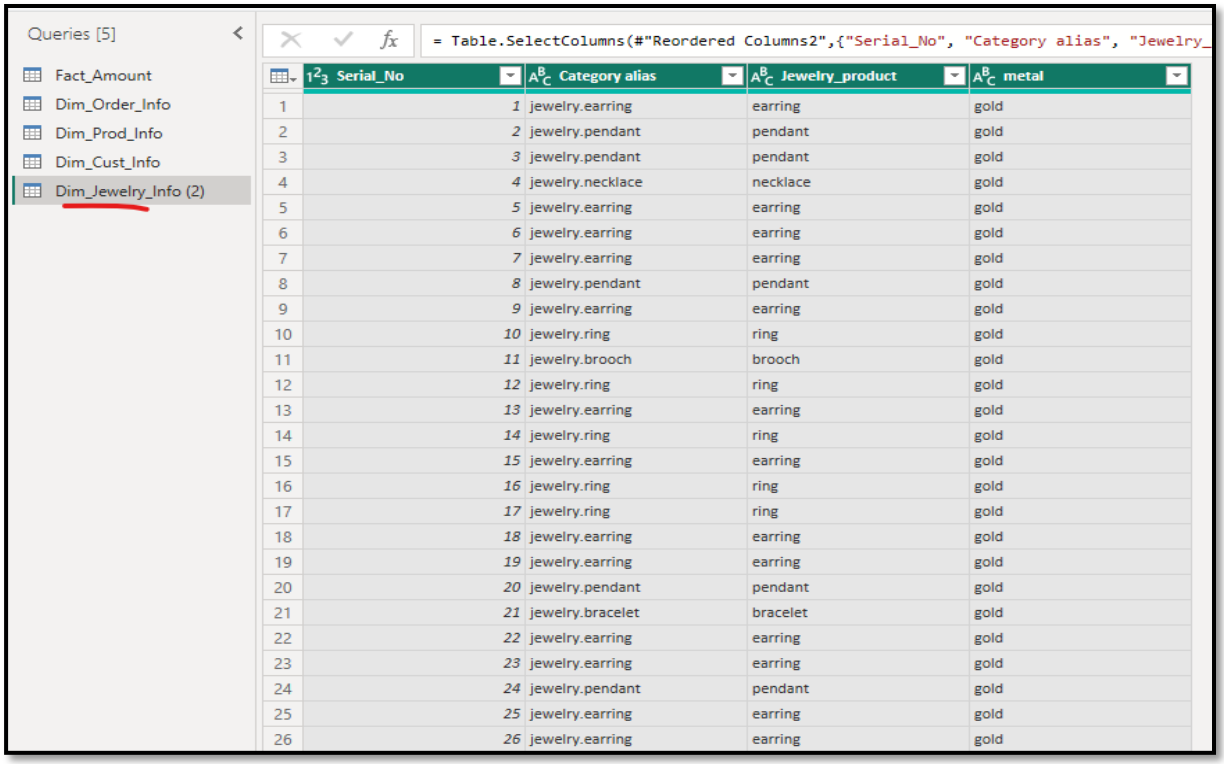


Figure-54: Dimension Dim_ Jewelry_Info

= Table.SelectColumns(#"Reordered Columns2",{"Serial_No", "Category alias", "Jewelry_product", "metal"})

Fact Fact_Amount

The “Fact_Amount” table contains the following columns-

a) Serial_No
b) Price in USD
c) Quantity of SKU in the order

Serial_No	Quantity of SKU in the order	Price in USD	Brand ID
1	1	561.51	0
2	1	54.66	1
3	1	88.9	0
4	1	417.67	0
5	1	102.27	1
6	1	150.55	0
7	1	83.42	0
8	1	60.27	2
9	1	217.4	1
10	1	180.71	1
11	1	232.1	1
12	1	203.92	1
13	1	84.45	0
14	1	254.45	0
15	1	251.15	0
16	1	154.88	1
17	1	154.88	1
18	1	2171.1	0
19	1	102.6	1
20	1	61.51	1
21	1	335.48	0

Figure-55: Creation of Fact_Amount Table

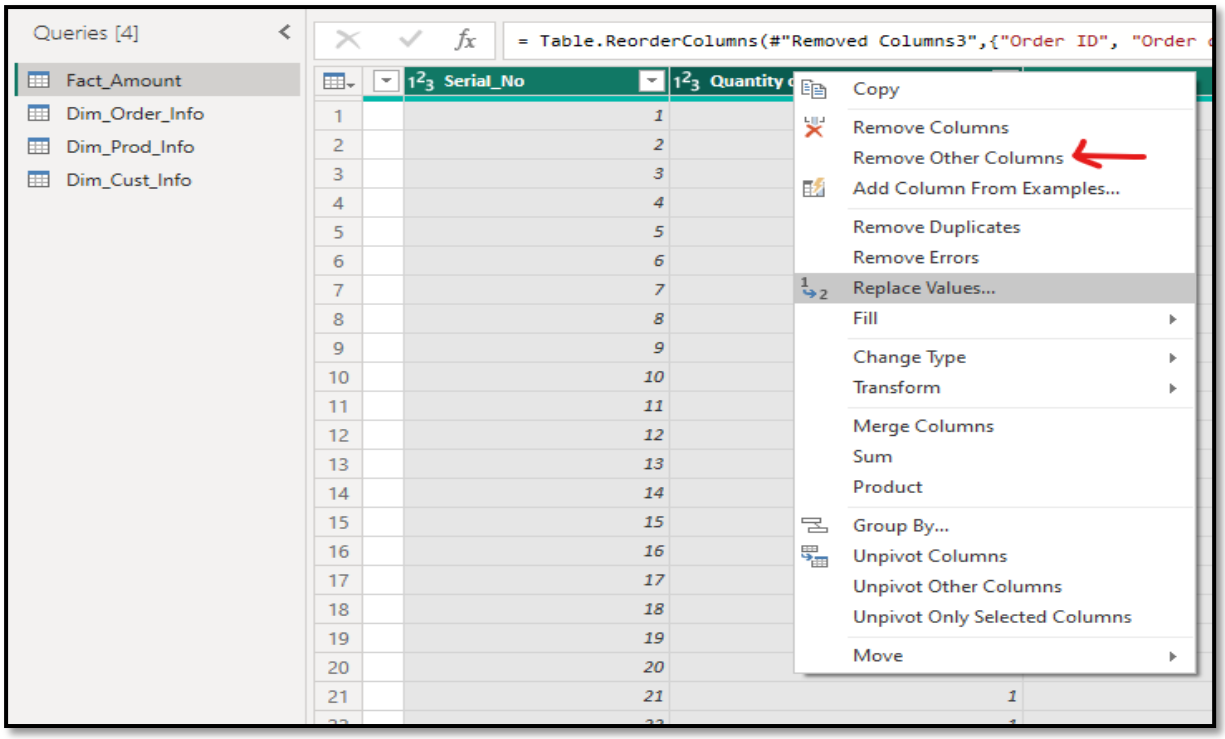
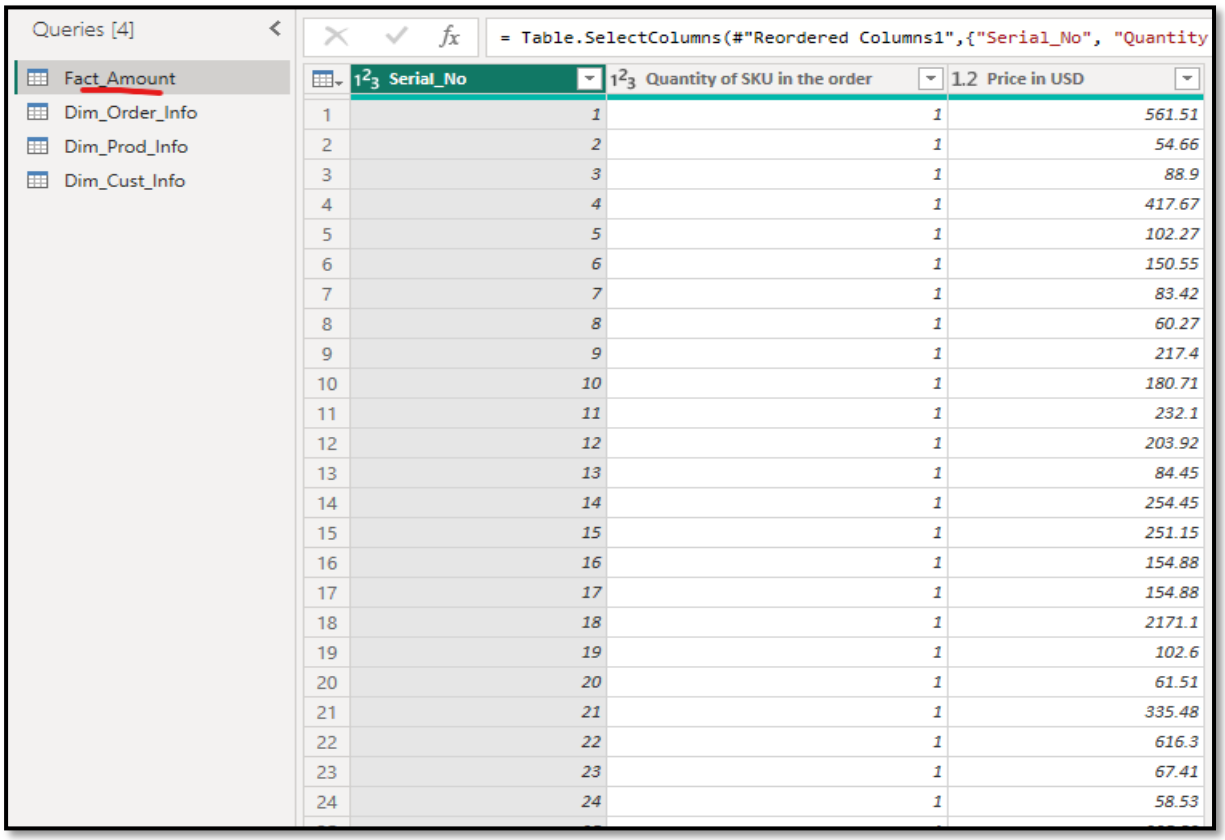


Figure-56: Fact Fact_Amount Table (Removing extra column)



	Serial_No	Quantity of SKU in the order	Price in USD
1	1	1	561.51
2	2	1	54.66
3	3	1	88.9
4	4	1	417.67
5	5	1	102.27
6	6	1	150.55
7	7	1	83.42
8	8	1	60.27
9	9	1	217.4
10	10	1	180.71
11	11	1	232.1
12	12	1	203.92
13	13	1	84.45
14	14	1	254.45
15	15	1	251.15
16	16	1	154.88
17	17	1	154.88
18	18	1	2171.1
19	19	1	102.6
20	20	1	61.51
21	21	1	335.48
22	22	1	616.3
23	23	1	67.41
24	24	1	58.53

Figure-57: Fact_Amount Table

= Table.SelectColumns(#"Reordered Columns1",{"Serial_No", "Quantity of SKU in the order", "Price in USD"})

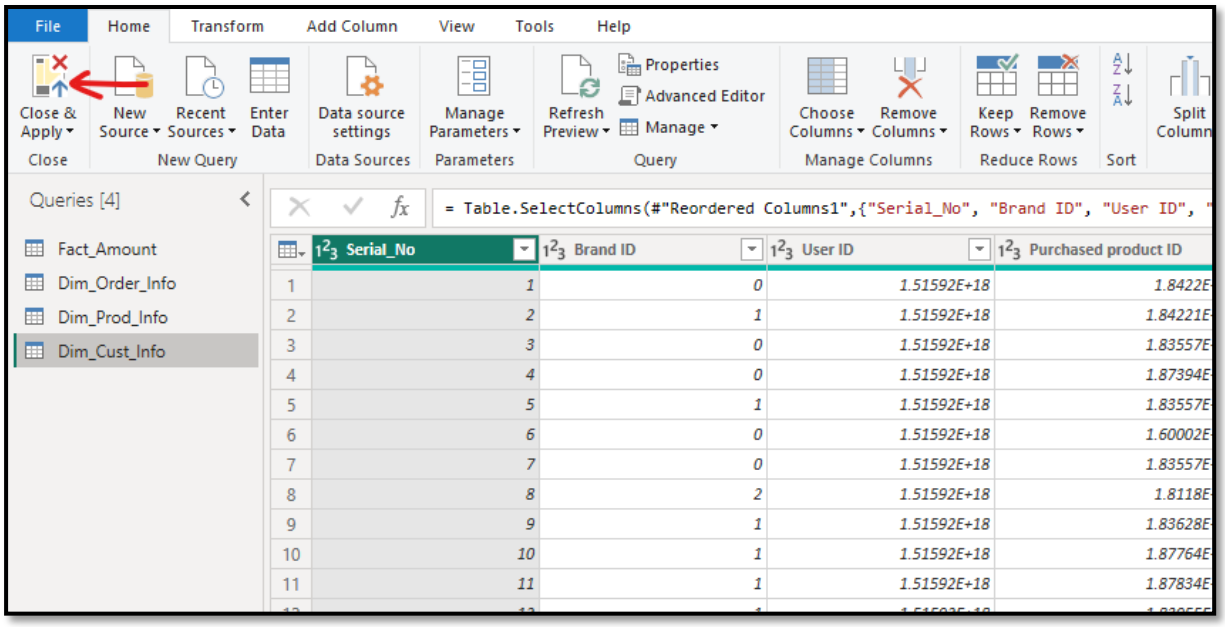


Figure-58: Saving of Dimension and Fact Table

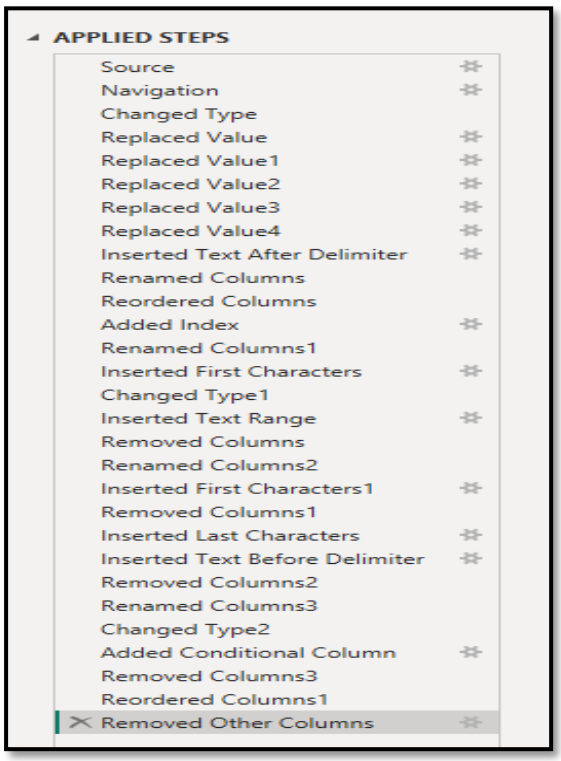


Figure-59: Applied step of data transformation

Figure-60: Loading of Dimension and Fact Table

List of Fact and Dimensions Tables

The list of all fact and dimension tables can be seen in right-top side of the power-bi screen. Here in the following image, the list of fact and dimension can be seen.

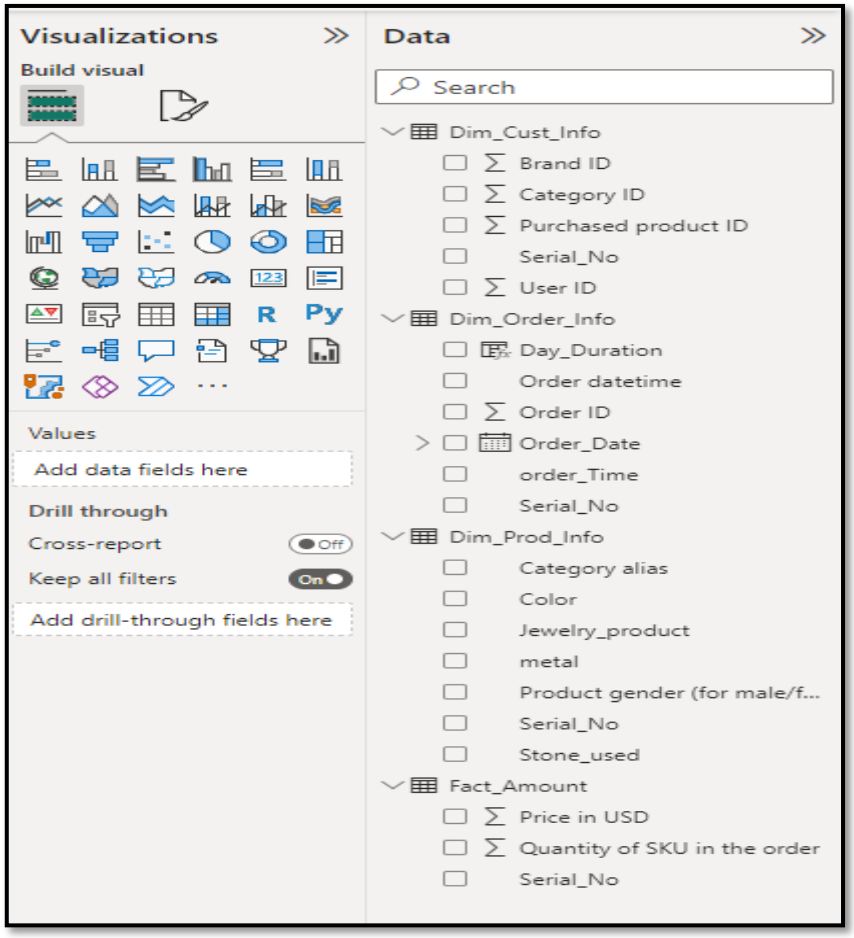


Figure-61: List of fact and dimensions

Data models and relationships

To visualize the data in power-bi, first the relationship between the fact and dimension should be developed.

The relationship developed here, is based on the “Star-schema”.

Star Schema [1:1 relationship]

- This design simplifies querying and reporting, making it easier for analysts to retrieve and analyze information.
- Fact table is in center and surrounded by multiple dimension tables.

Manage Relationship

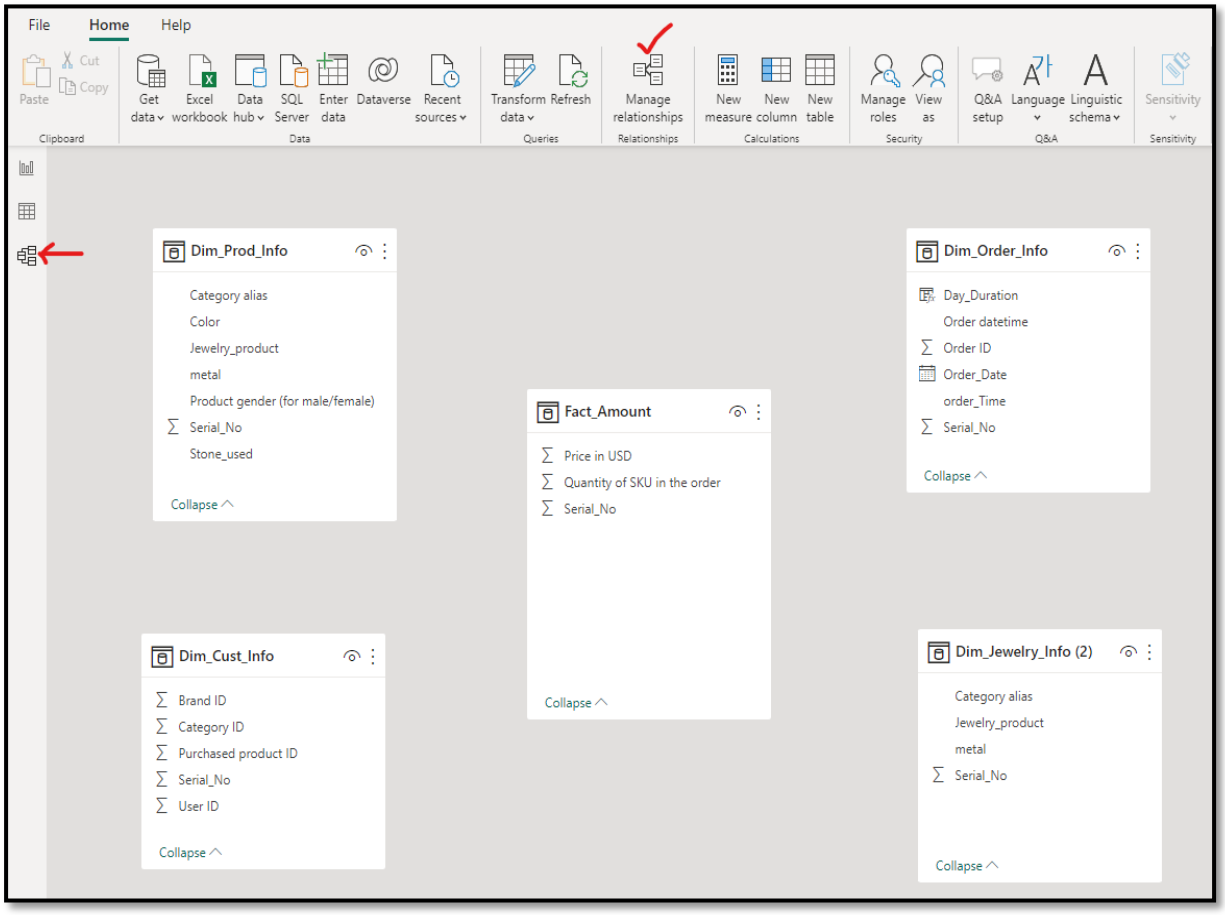


Figure-62: Manage Relationship Option

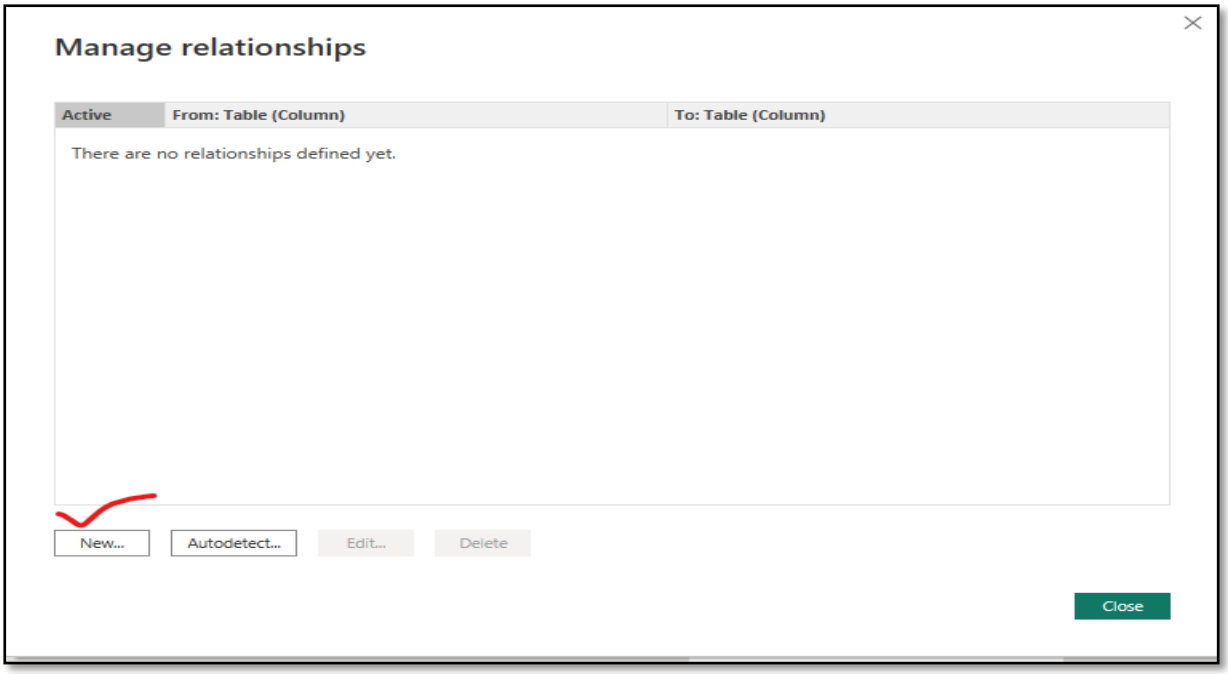


Figure-63: Manage Relationship pop up

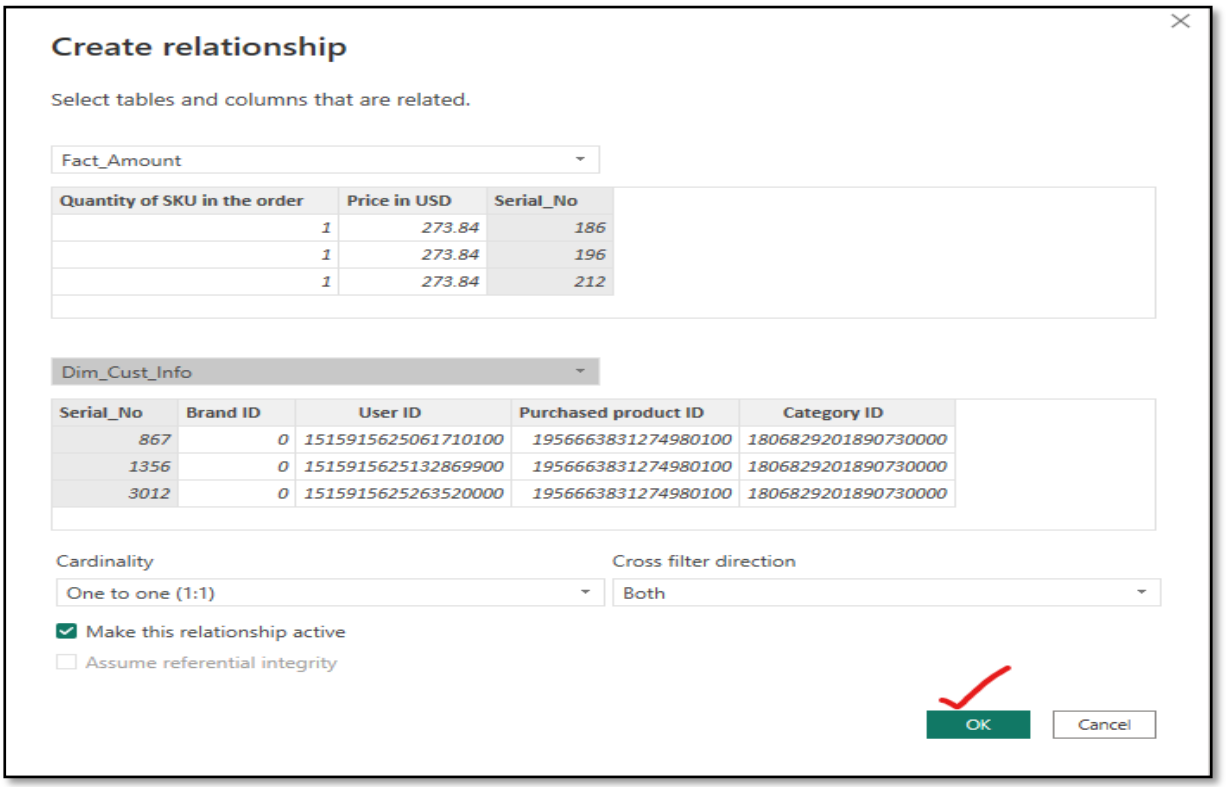


Figure-64: Create Relationship b/w Fact_Amount & Dim_Cust_Info

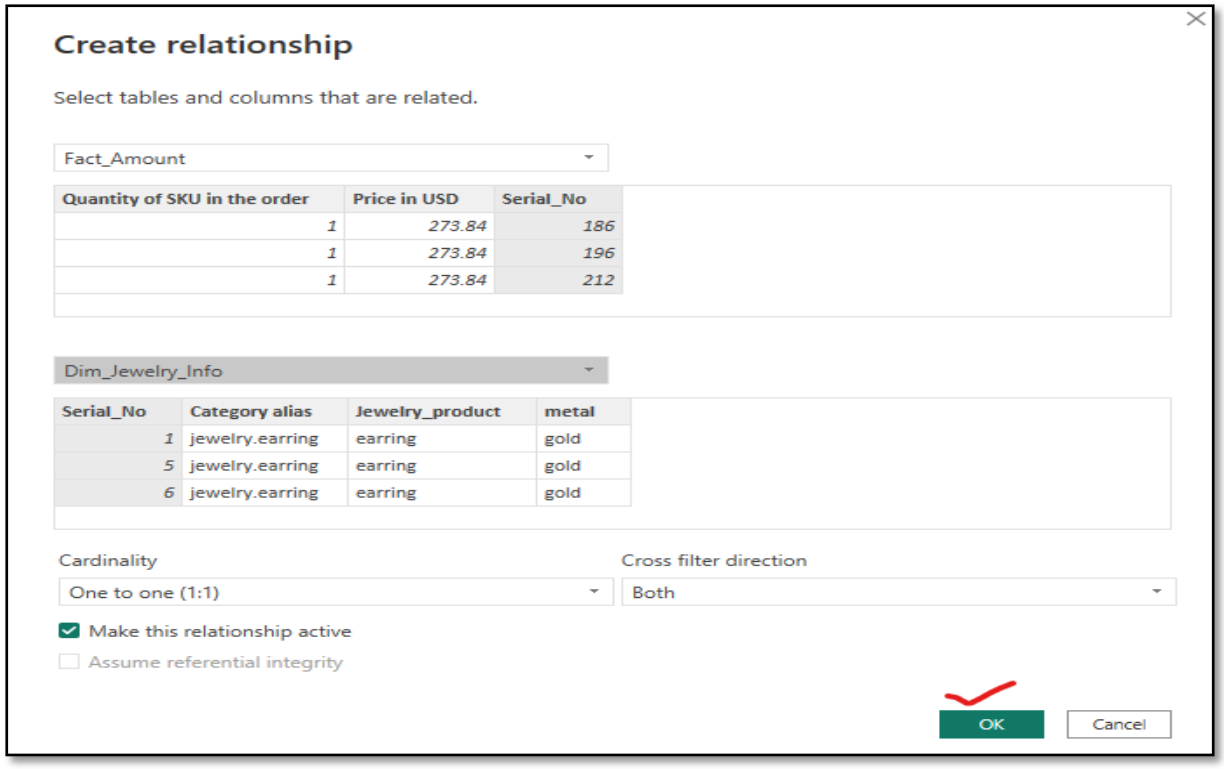


Figure-65: Create Relationship b/w Fact_Amounts & Dim_Jewelry_Info

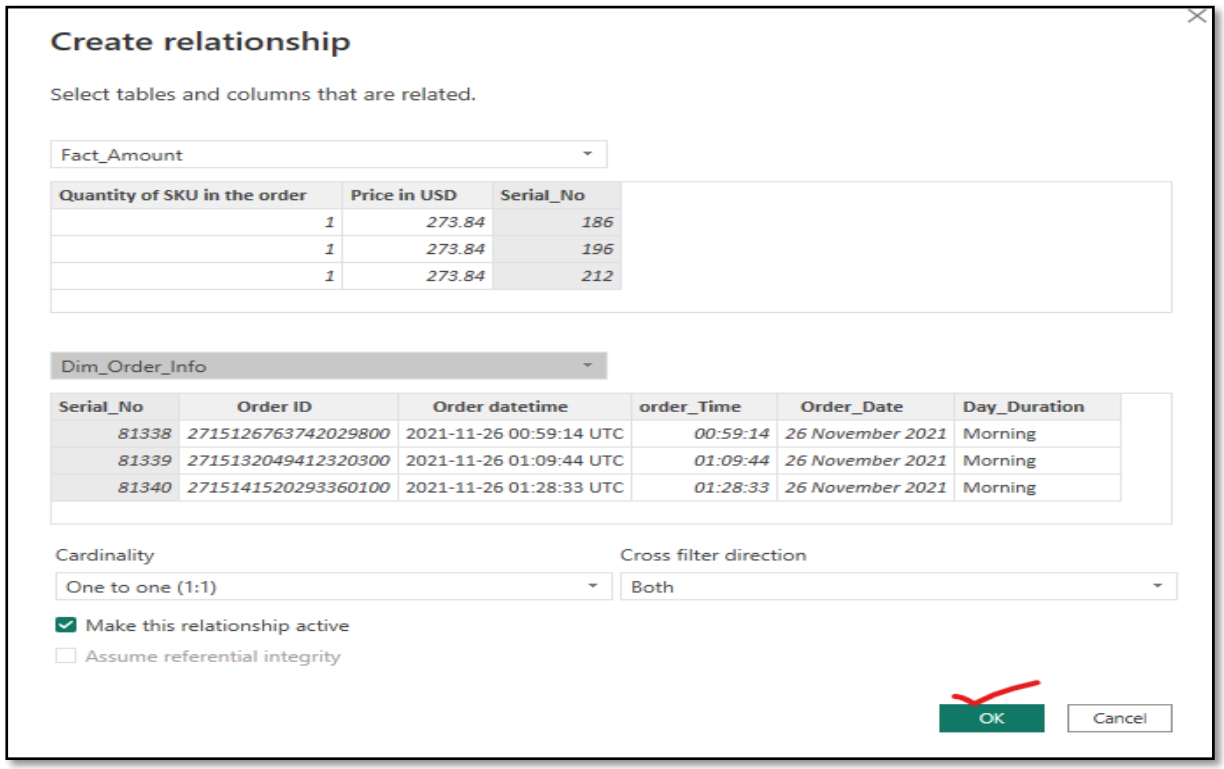


Figure-66: Create Relationship b/w Fact_Amounts & Dim_Order_Info

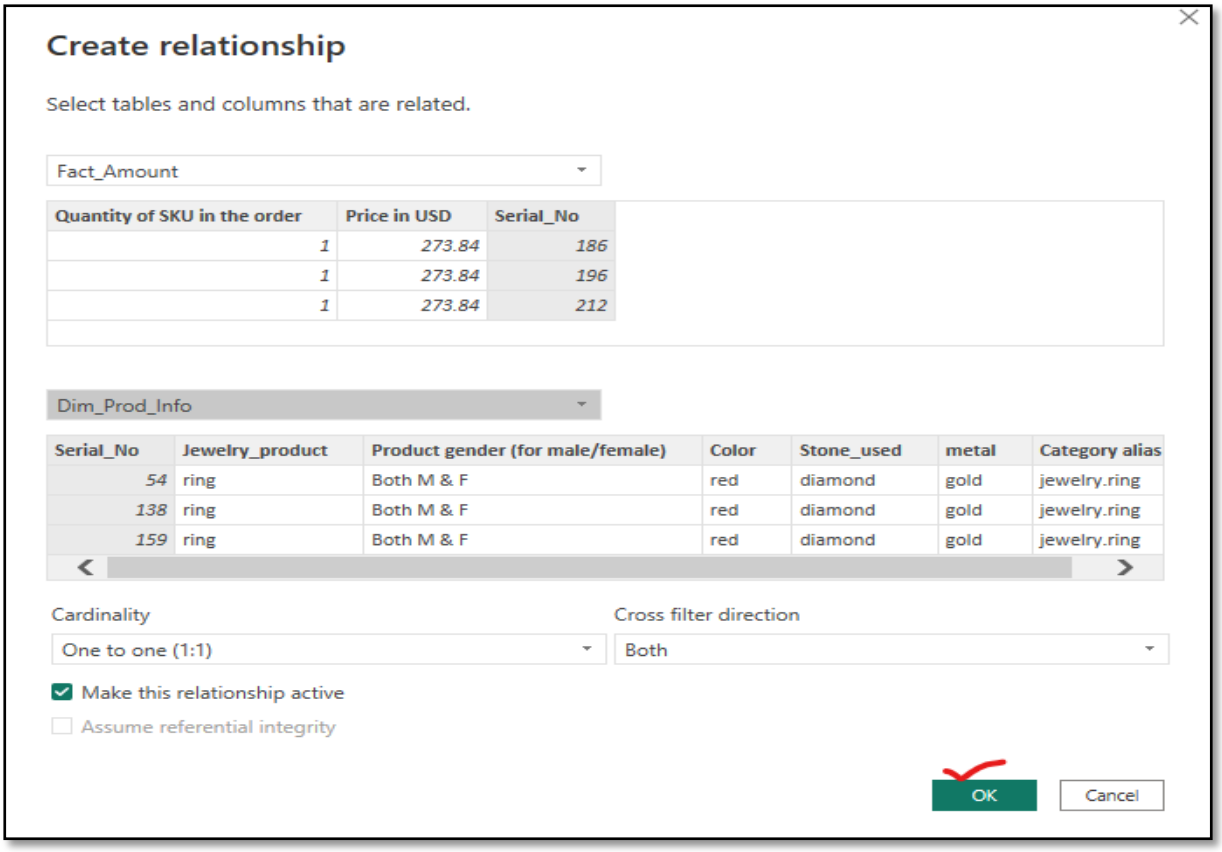


Figure-67: Create Relationship b/w Fact_Amount & Dim_Prod_Info

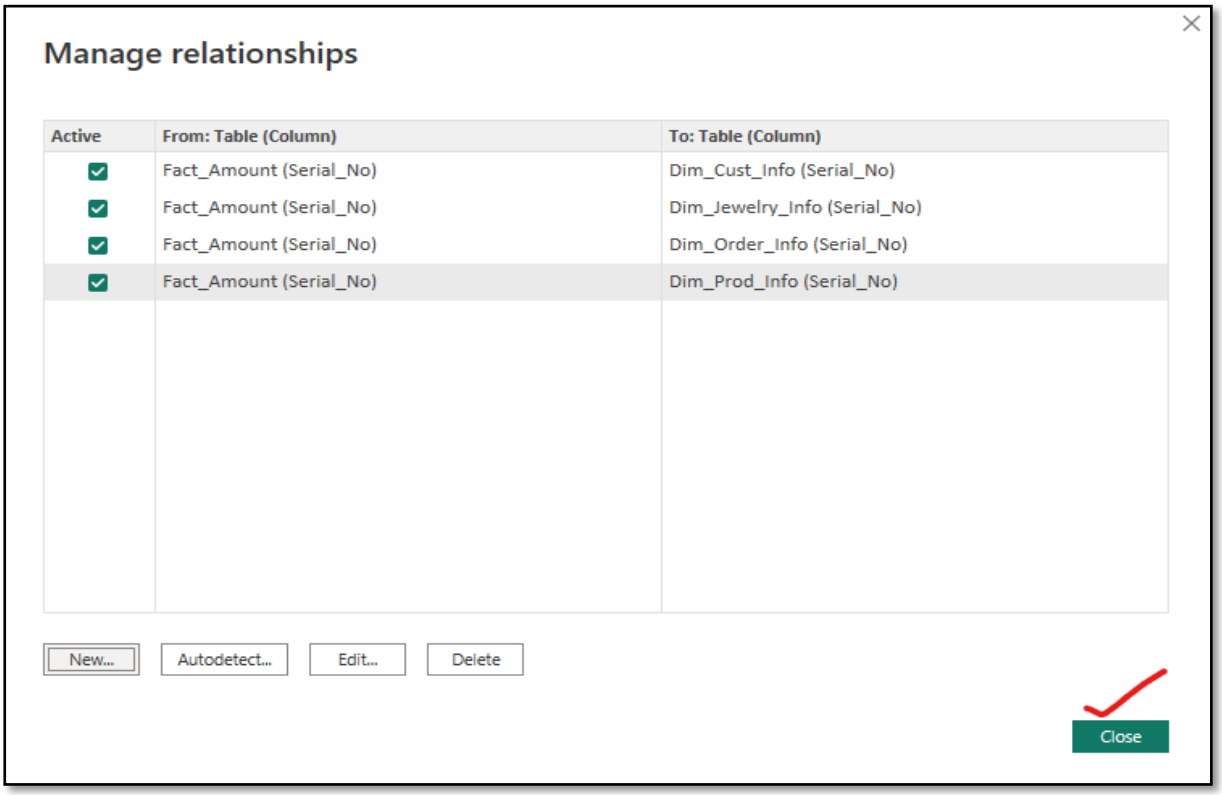


Figure-68: View of Relationship

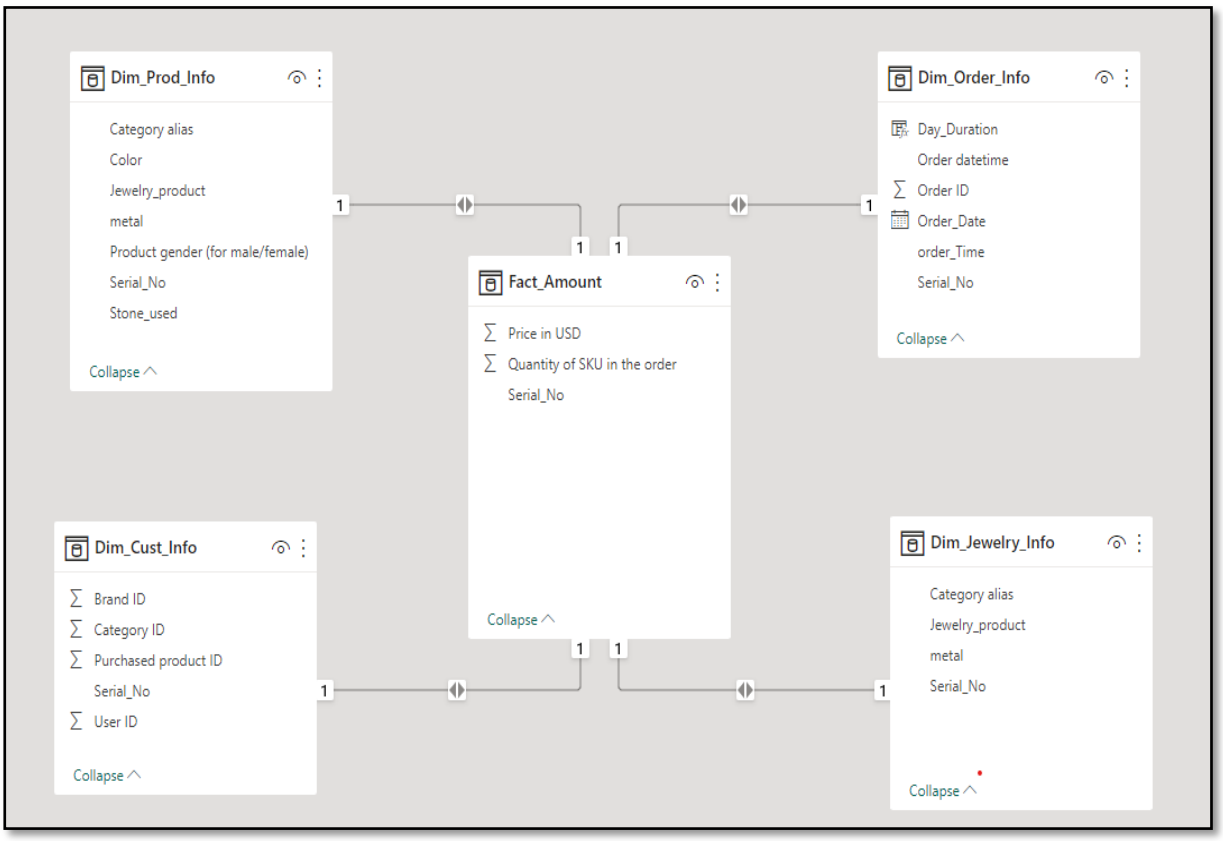


Figure-69: View of Relationship b/w Fact and Dimension Table

Data Visualization

Adding Canvas Background

- In Power BI, the canvas background refers to the background area of a report or dashboard where visualizations, charts, and other elements are placed.
- The canvas is the space where you design and arrange your data visualizations to convey insights to your audience.

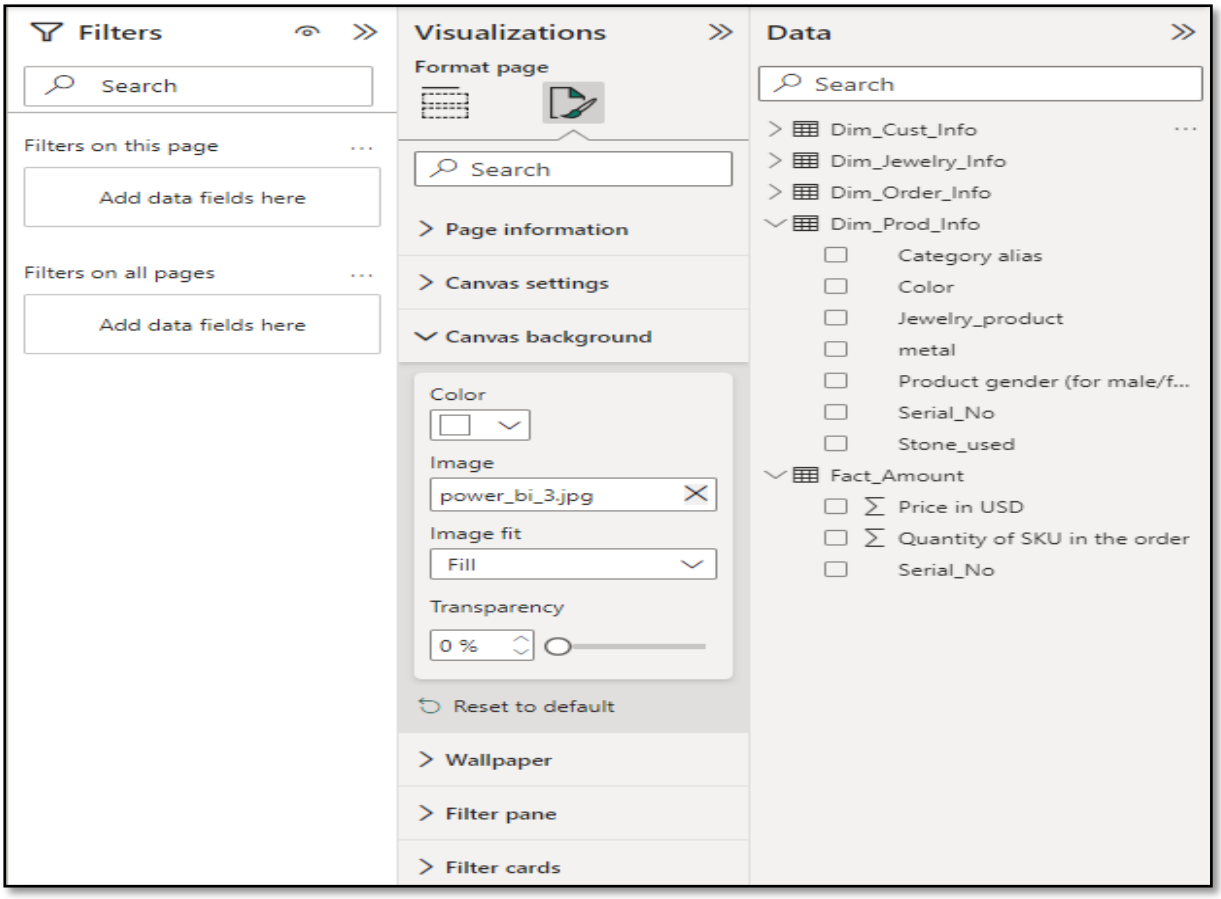


Figure-70: Format of Canvas Background

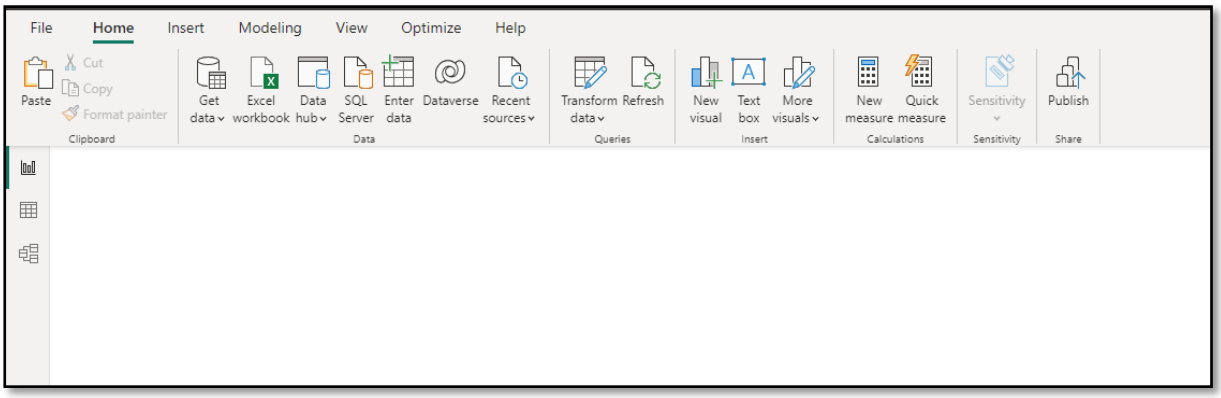


Figure-71: View of Canvas Background

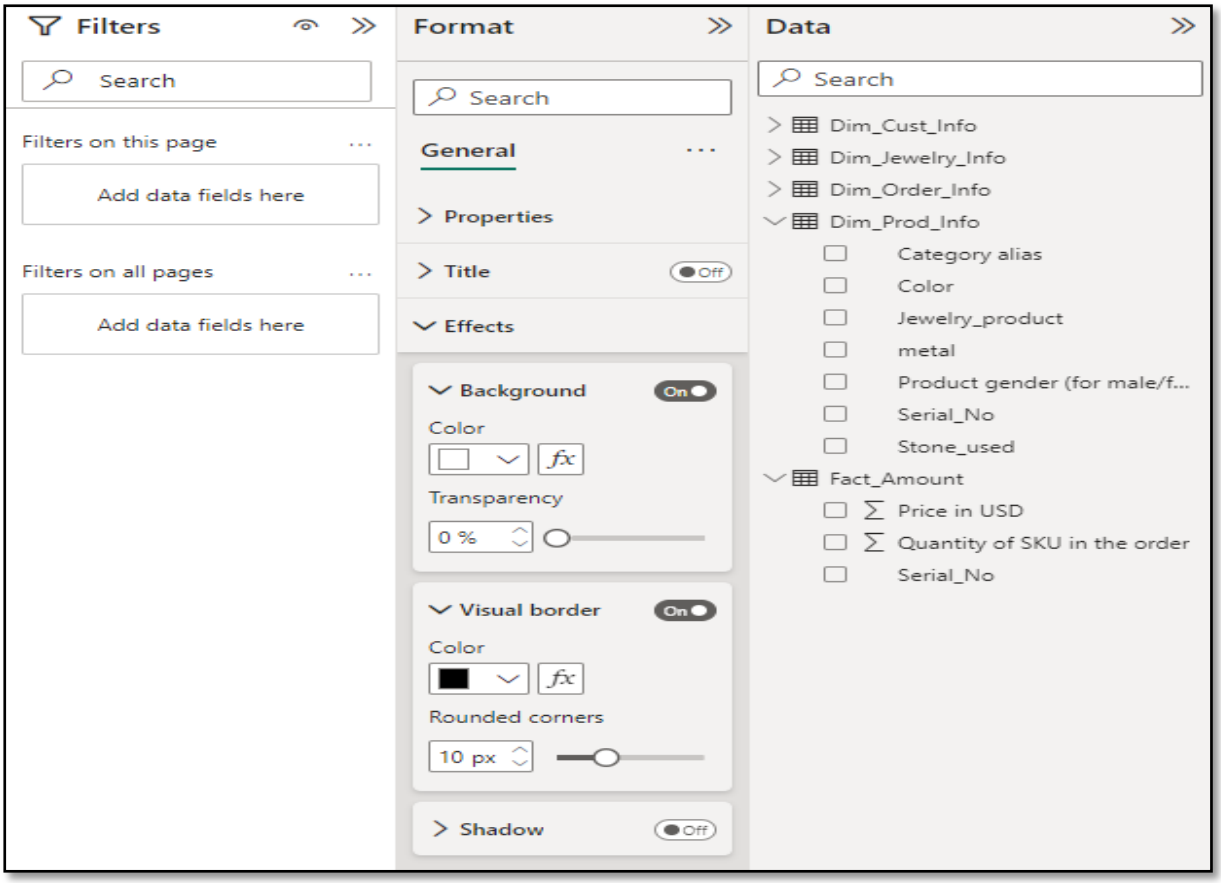


Figure-72: Format of Text Box

Insert label cards

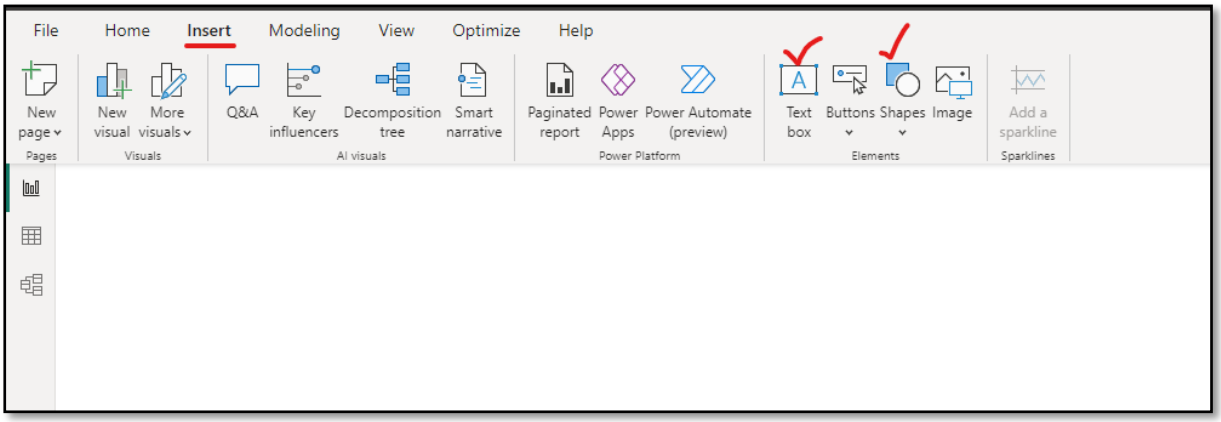


Figure-73: Label-Card Option

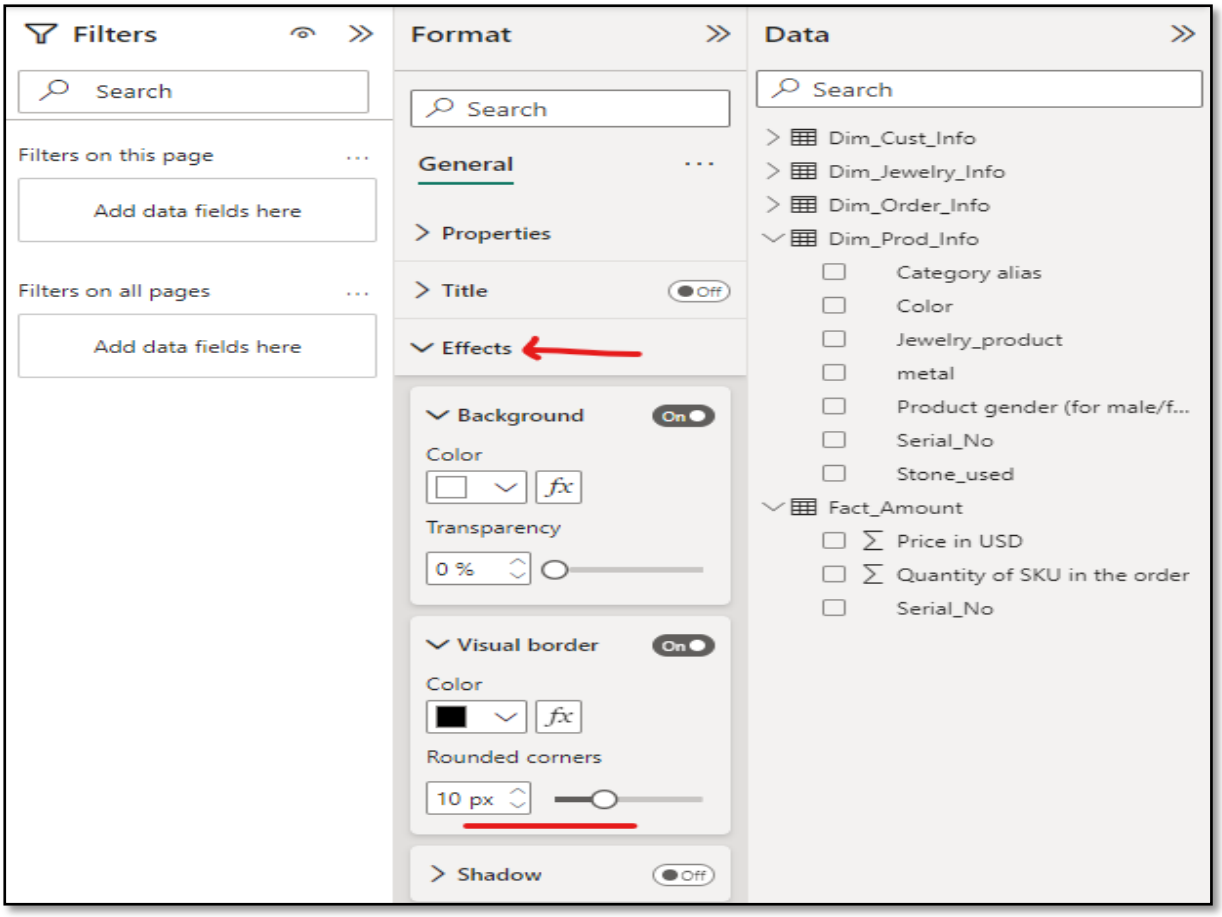


Figure-74: Format of Labels

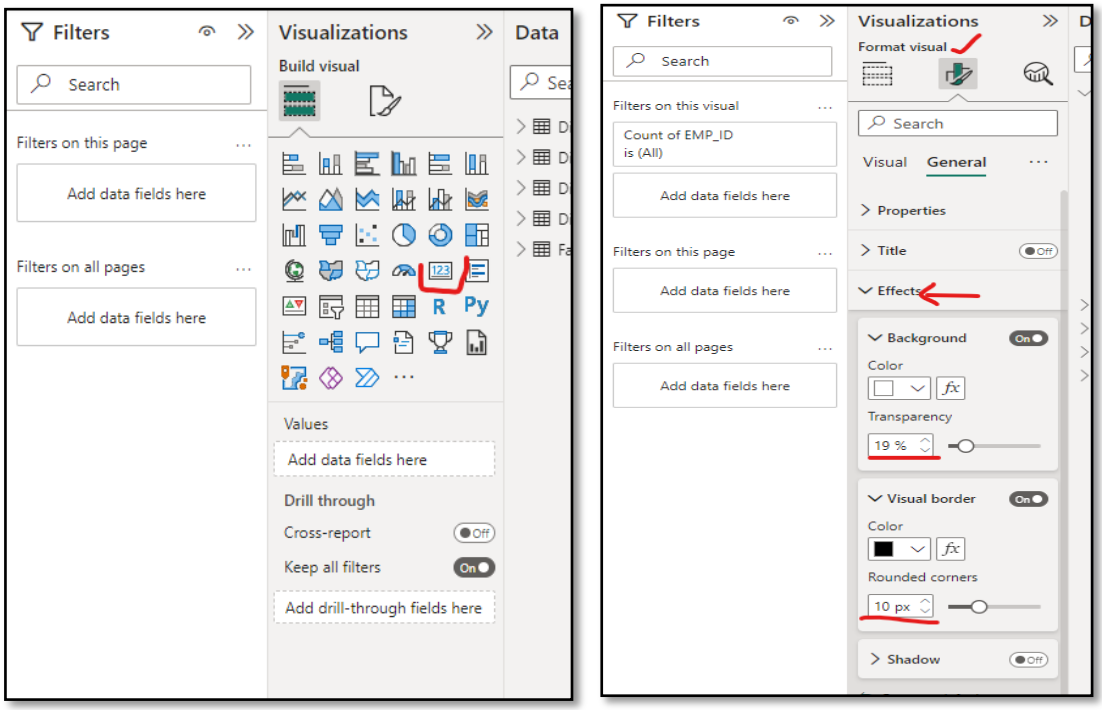


Figure-75: Format & Icon of Card

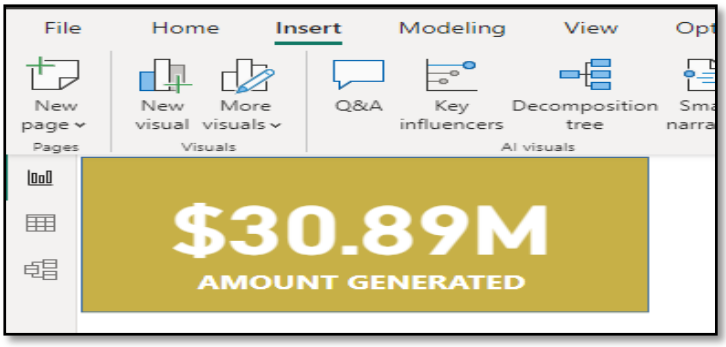


Figure-76: View of Template

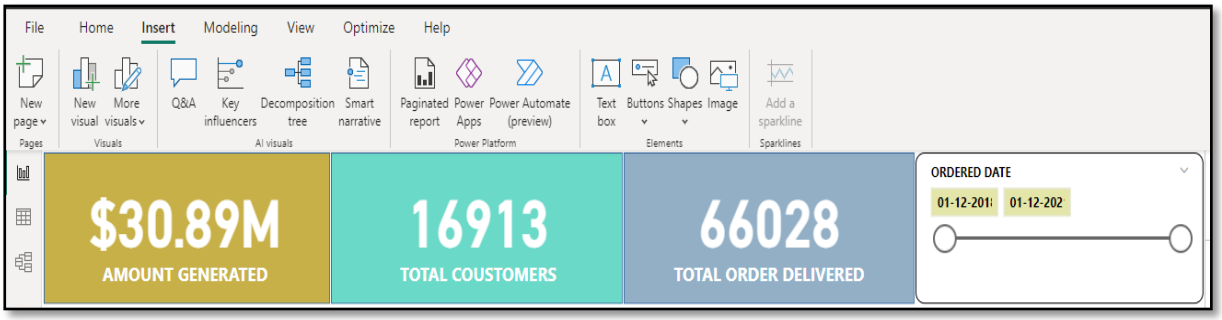


Figure-77: View of Label-Card 1

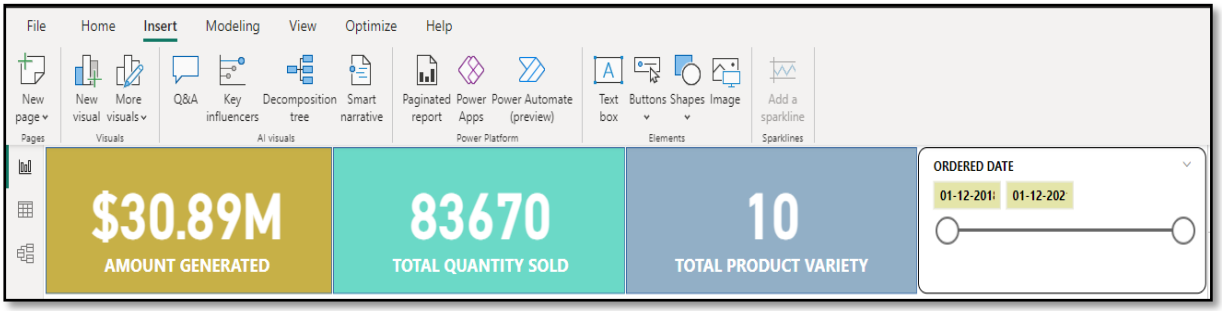


Figure-78: View of Label-Card 2

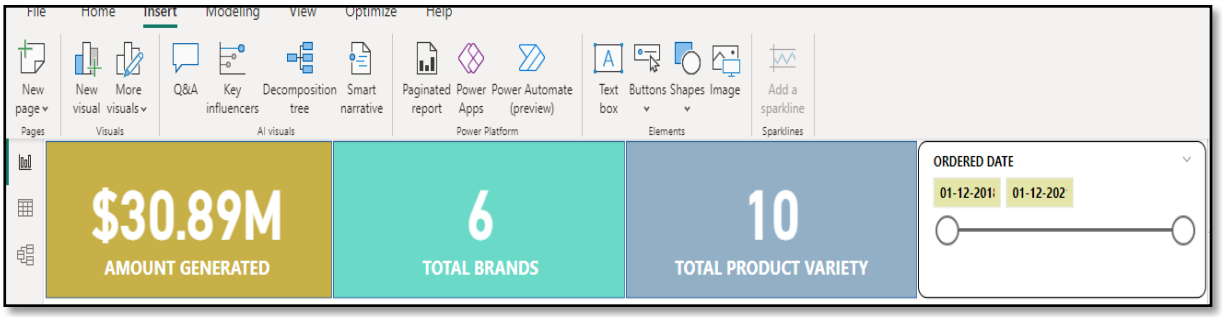


Figure-79: View of Label-Card 3

Buttons and bookmarks

The button on the analysis page is given to switch between the key influencer and decomposition tree.

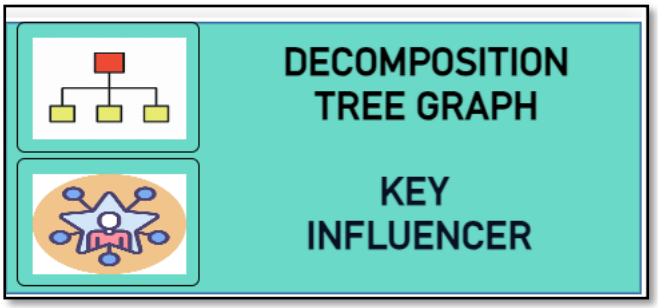


Figure-80: View of buttons

The following bookmark format is set to display between both.

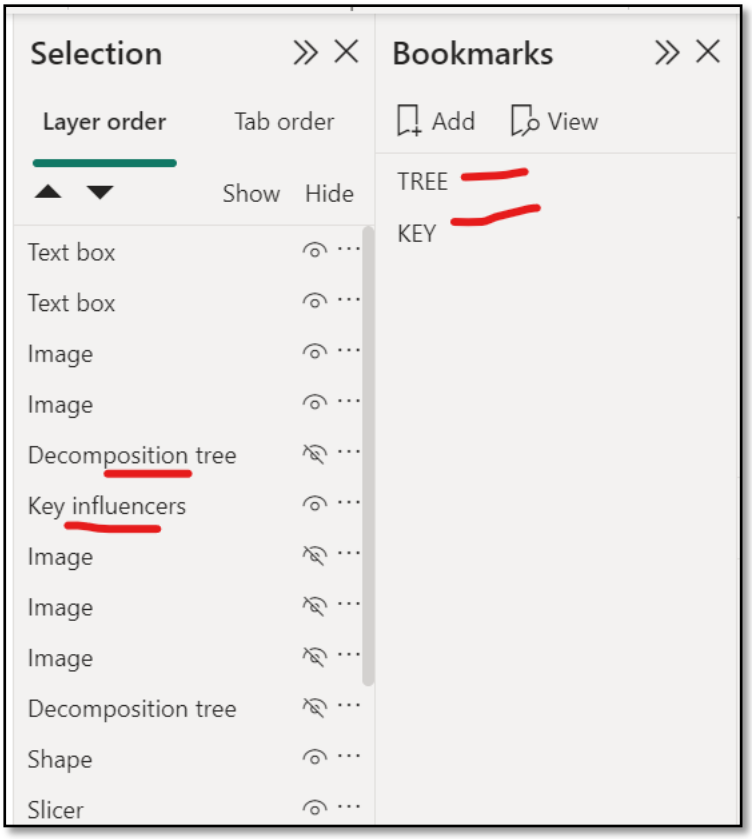


Figure-81. View of Bookmarks

Dashboard

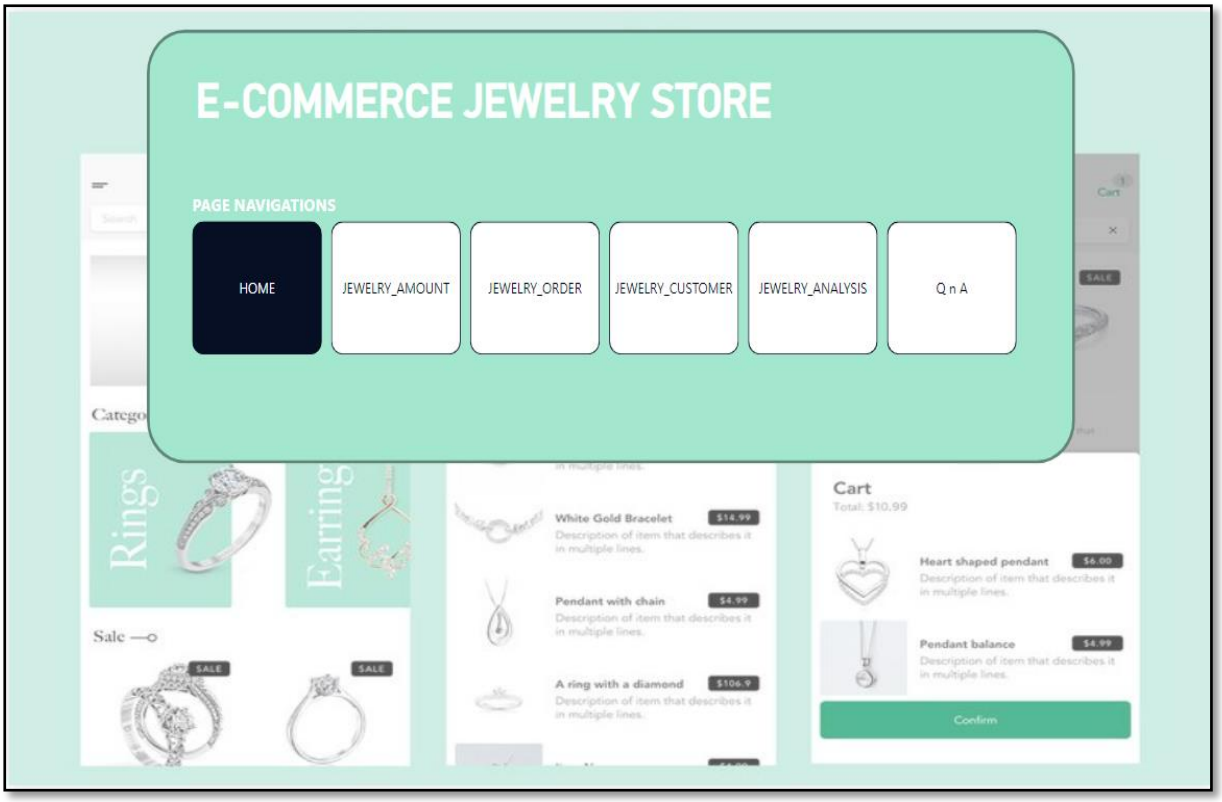


Figure-82. View of Home page of Dashboard

The home page navigation button has following buttons:

1. HOME
2. JEWELRY_AMOUNT
3. JEWELRY_ORDER
4. JEWELRY_CUSTOMER
5. JEWELRY_ANALYSIS
6. Q n A

[END]