



07/01/2025

BIG DATA AND BUSINESS INTELLIGENCE

Module Leader: Annalisa Occhipinti

Module Code: CIS4008-N

Submission Date: 07/01/2025

STUDENT NAME: USAMA MASOOD

STUDENT NUMBER: S3355014

Contents

A. EXECUTIVE SUMMARY	3
Introduction	Error! Bookmark not defined.
Introduction	3
Key Findings	4
Recommendations.....	4
B. INTRODUCTION	4
i) Data Source	4
ii) BI Questions	4
C. FINDINGS BASED ON ANALYSIS AND EVALUATION	5
1. What factors contribute to gold achieving significantly higher sales (₹32,694.05K) compared to silver (₹94.22K) and platinum (₹1.69K)?	5
What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?	5
What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?	6
What factors contribute to tourmaline achieving significantly higher sales (112,759) compared to spinel (18,350) and turquoise (16,503)?	6
What factors contribute to the significantly higher average price of jewelry rings (0.37K) compared to studs (0.3K) and souvenirs (0.04K)?	7
What factors contribute to the observed sales of gold (0.05K) compared to other materials, and what strategies can be implemented to optimize sales?	8
What factors contribute to female sales accounting for 50% of total sales, and how can this percentage be increased to achieve a more balanced contribution?	8
What factors contribute to jewellery necklaces and earrings having significantly higher average prices (approximately 400) compared to jewellery rings (approximately 300) and pendants (approximately 200)?	8
What factors contribute to the observed average prices of jewellery earrings (0.27K) compared to jewellery rings (0.2K)?	9
What factors contribute to amethyst achieving significantly higher total sales (₹5,82,323.07) compared to other stones such as amber (₹68,873.80) and agate (₹14,592.22)?	9
APPENDIX: BI DESIGN	13
I. DATA PRE-PROCESSING AND DATA CLEANING	13
Load and Clean Dataset	13
Connecting to Power BI	13
Transformed the data	14

Data Processing:	14
Data Modelling:	23
Create Measures and Calculated Columns	24
Dashboard Visualizations:	25
Home page.	25

A. EXECUTIVE SUMMARY

Introduction

This report has been prepared for a jewelry shop where vital key performance metrics shall be shown with the application of Power BI. This business belongs to a jeweler domain; with dynamically challenging high-speed competitors, many organizations can trace diverse functions executed inside their outlet store. Some of the examples are a trend of selling, customer buying behavior, control of inventory in store, and store profitability. This report utilizes the robust power of data visualization and business acumen of Power BI to facilitate insightful analysis with actionable recommendations aimed at optimizing performance and driving growth in the jewelry store. Such data will include sales transactions, demographics of customers, and inventory levels among others which may be necessary for business operations. The Power BI visualizations that will be generated will offer an overview of how the store has performed, making it possible for the decision makers to identify the pattern, determine trends, and come up with proper decisions towards strategic planning.

For any other category of sales will be offered so that users may get to dig into the real sales by specific categories, by profit margins customer demographics, the cycle of inventory turnover, and similar other indicators so that the essential management of a jewelry store through dynamic dashboards enhance operational efficiency increase profitability while can provide a customer experience.

Key Findings

Recommendations

B. INTRODUCTION

i) Data Source

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

ii) BI Questions

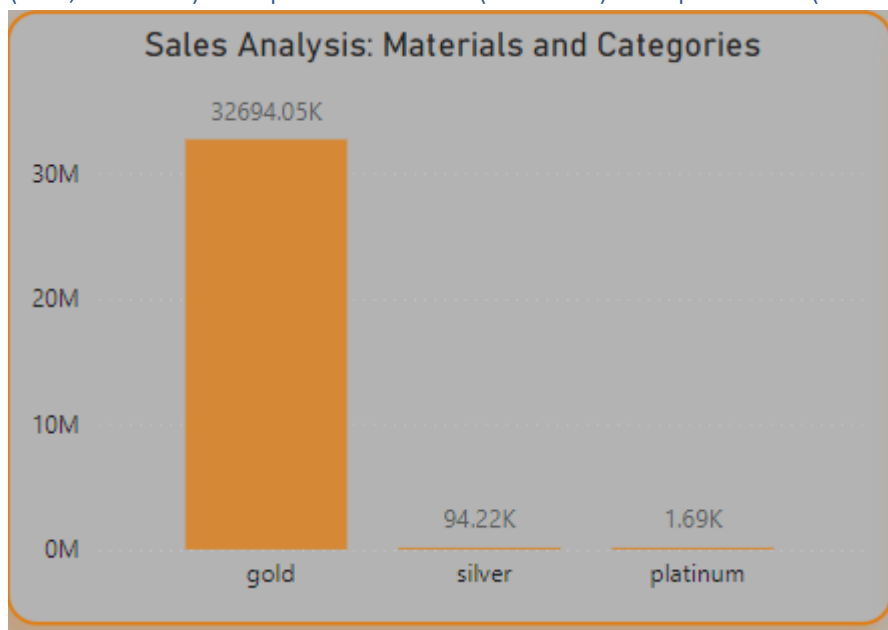
BI Questions:

- 1.What factors contribute to gold achieving significantly higher sales (₹32,694.05K) compared to silver (₹94.22K) and platinum (₹1.69K)?
2. What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?
- 3.What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?
- 4.What factors contribute to tourmaline achieving significantly higher sales (112,759) compared to spinel (18,350) and turquoise (16,503)?
- 5.What factors contribute to the significantly higher average price of jewellery rings (0.37K) compared to studs (0.3K) and souvenirs (0.04K)?
- 6.What factors contribute to the observed sales of gold (0.05K) compared to other materials, and what strategies can be implemented to optimize sales?
- 7.What factors contribute to female sales accounting for 50% of total sales, and how can this percentage be increased to achieve a more balanced contribution?
- 8.What factors contribute to jewellery necklaces and earrings having significantly higher average prices (approximately 400) compared to jewellery rings (approximately 300) and pendants (approximately 200)?
- 9.What factors contribute to the observed average prices of jewellery earrings (0.27K) compared to jewellery rings (0.2K)?

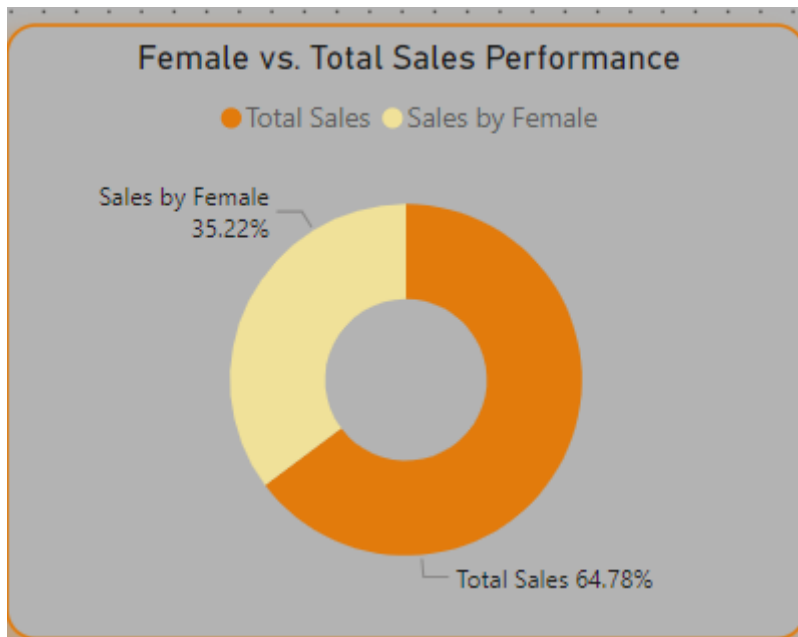
10. What factors contribute to amethyst achieving significantly higher total sales (₹5,82,323.07) compared to other stones such as amber (₹68,873.80) and agate (₹14,592.22)?

C. FINDINGS BASED ON ANALYSIS AND EVALUATION

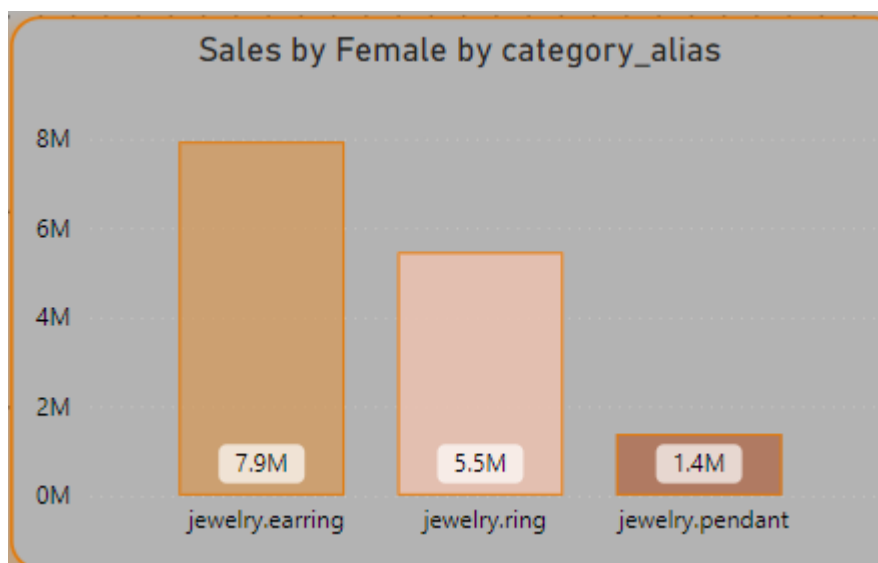
1. What factors contribute to gold achieving significantly higher sales (₹32,694.05K) compared to silver (₹94.22K) and platinum (₹1.69K)?



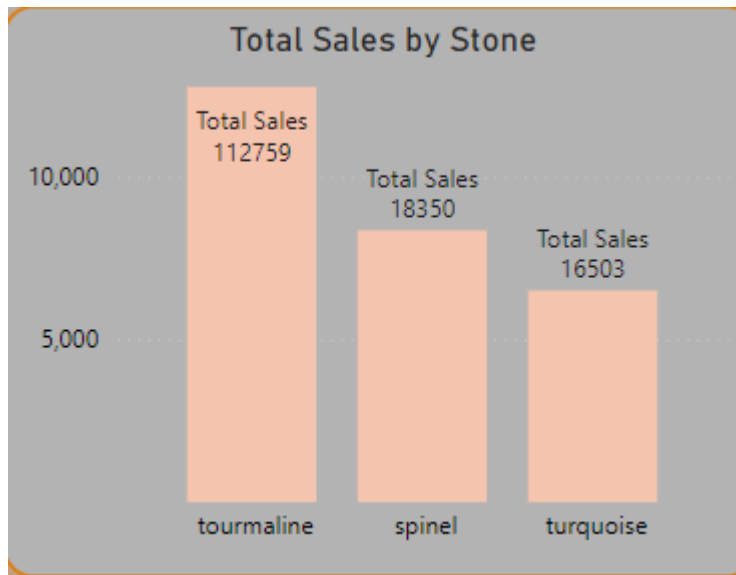
What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?



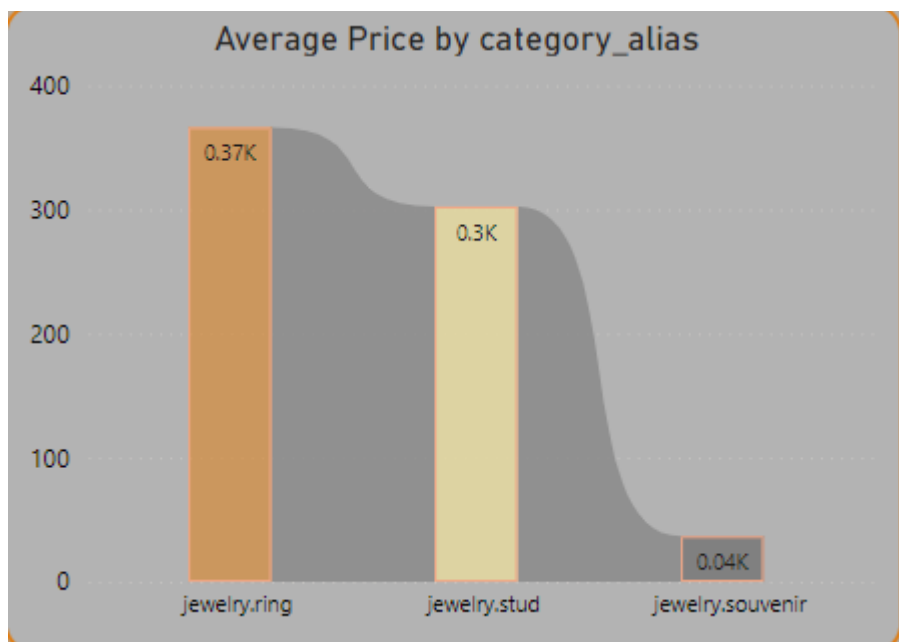
What strategies can be implemented to increase female sales (currently 35.22% of total sales) to achieve a more balanced contribution compared to total sales (64.78%)?



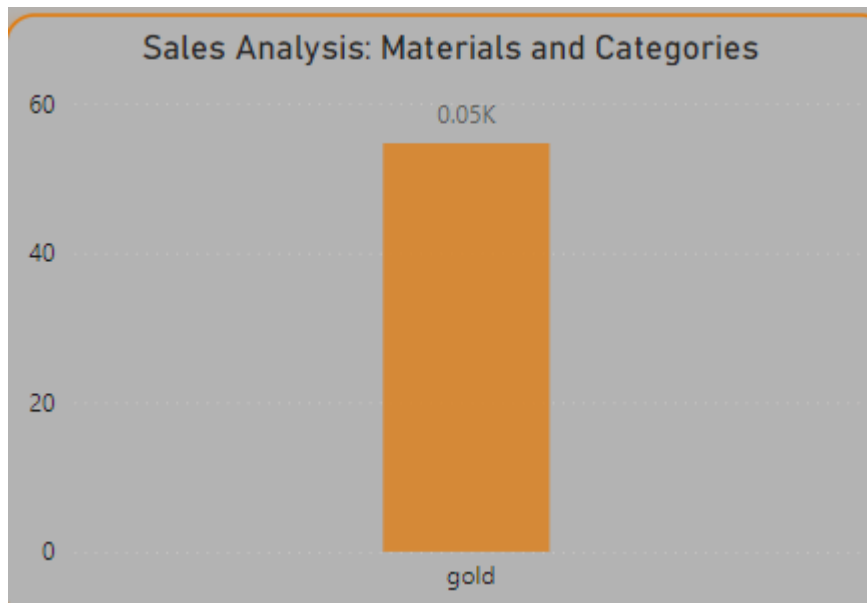
What factors contribute to tourmaline achieving significantly higher sales (112,759) compared to spinel (18,350) and turquoise (16,503)?



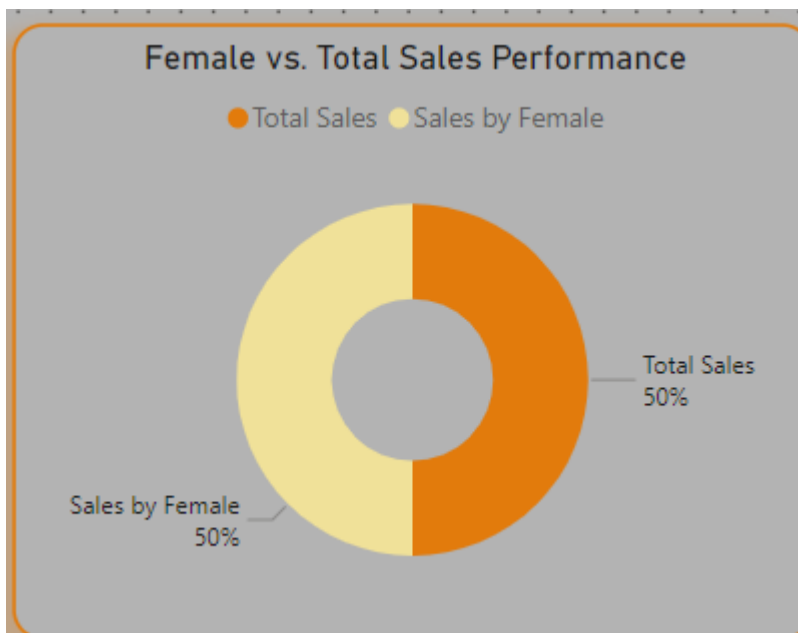
What factors contribute to the significantly higher average price of jewelry rings (0.37K) compared to studs (0.3K) and souvenirs (0.04K)?



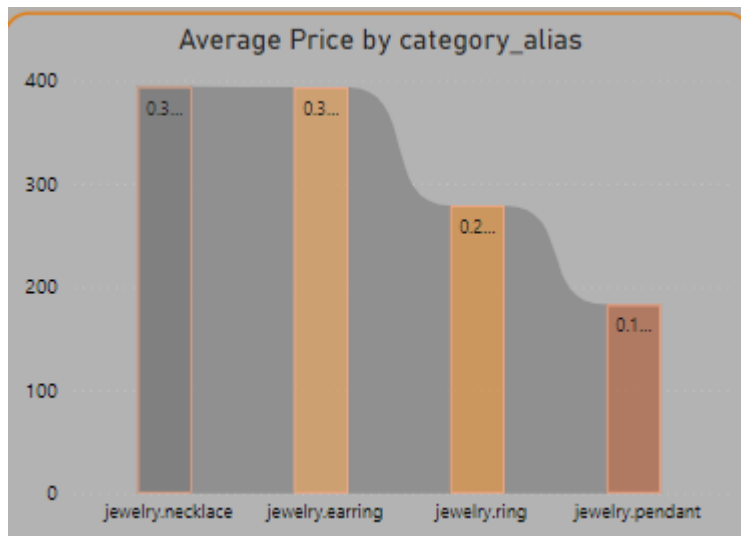
What factors contribute to the observed sales of gold (0.05K) compared to other materials, and what strategies can be implemented to optimize sales?



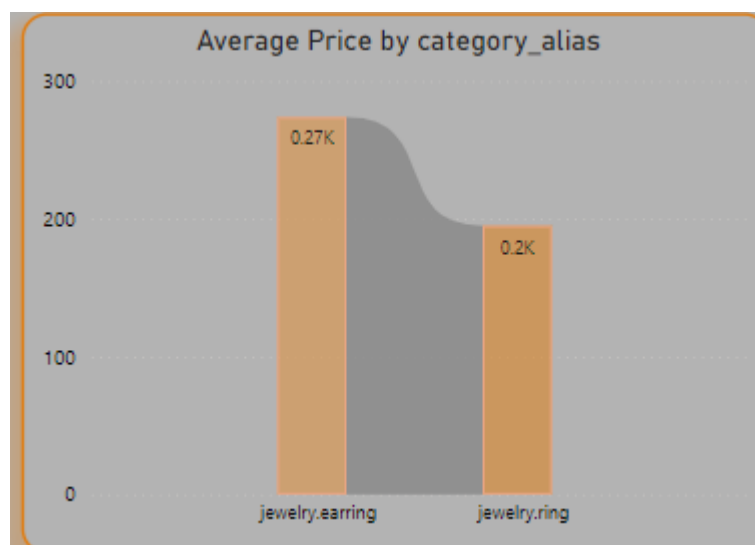
What factors contribute to female sales accounting for 50% of total sales, and how can this percentage be increased to achieve a more balanced contribution?



What factors contribute to jewellery necklaces and earrings having significantly higher average prices (approximately 400) compared to jewellery rings (approximately 300) and pendants (approximately 200)?

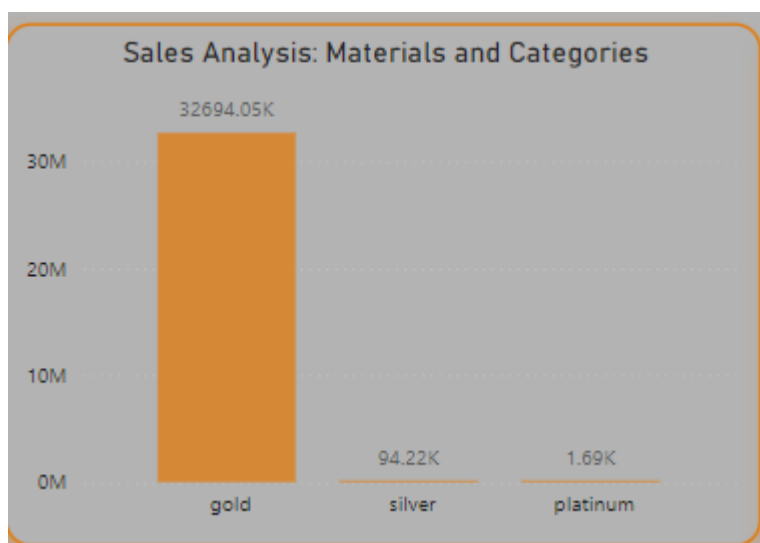


What factors contribute to the observed average prices of jewellery earrings (0.27K) compared to jewellery rings (0.2K)?



What factors contribute to amethyst achieving significantly higher total sales (₹5,82,323.07) compared to other stones such as amber (₹68,873.80) and agate (₹14,592.22)?

Material	Stone	Total Sales	Total Orders	Sales by Female	Sales by Male	Sales by Category	Color
gold	agate	14,592.22	19	14,459.38		14,592.22	red
gold	agate	1,401.92	2	1,401.92		1,401.92	white
gold	amber	68,873.80	184	14,150.08		68,873.80	red
gold	amethyst	5,82,323.07	1357	3,64,191.93		5,82,323.07	red
gold	amethyst	2,031.56	11	1,812.92		2,321.98	white
gold	amethyst	9,793.65	29	8,257.02		9,793.65	yellow
gold	chrysolite	10,634.36	64	4,427.80		10,634.36	red
gold	chrysolite	1,186.23	10	1,097.23		1,186.23	yellow
Total		2,39,56,422.32	49109	1,49,61,602.83	38,309.38	2,39,58,502.13	



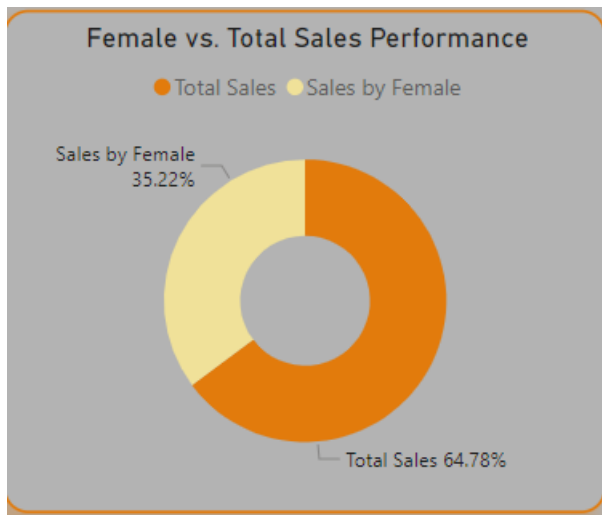
Insights from the Table (Sales Analysis: Materials and Categories):

- Gold has significantly higher sales (₹32,694.05K) compared to Silver (₹94.22K) and Platinum (₹1.69K).
- Silver and Platinum sales are considerably lower, with Platinum having the smallest share.

Questions:

Which material has the highest total sales, and by how much does it outperform the others?

Are there any trends in the data that suggest potential for increasing sales of Silver or Platinum?



Insights from the Donut Chart (Total Sales and sales by female):

- Female sales contribute 35.22% of the total sales and 64.78% represents the overall sales performance, with the balance from other sales or categories.

Customer Analysis:

Insights from the Line and Stacked column Chart (Category Alias and sales by female):

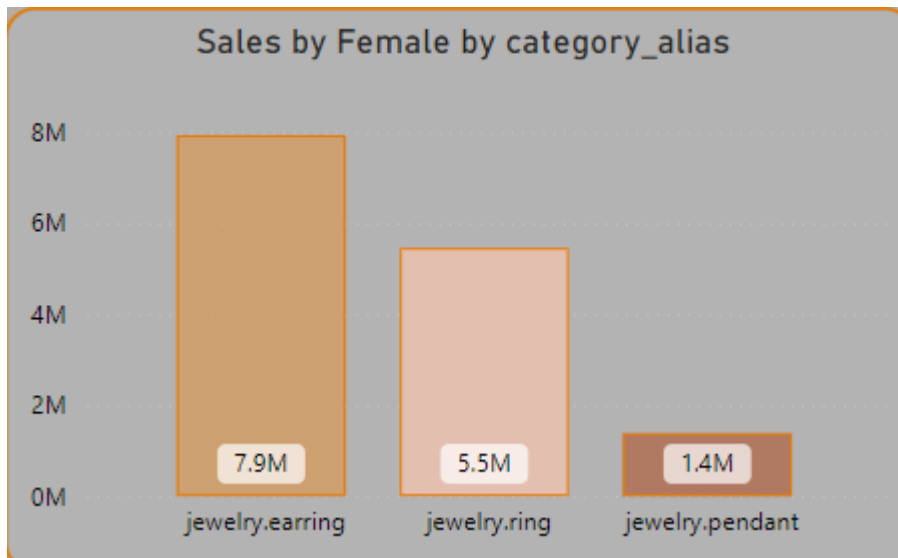
- Earrings lead with 7.9M in sales, showing strong performance among female customers.
- Rings follow with 5.5M in sales, indicating a significant contribution.
- Pendants have a smaller share, with 1.4M in sales, suggesting lower demand from female customers for this category.

Questions:

Why do earrings have the highest sales among female customers, and how can sales of pendants be improved?

What factors might be causing the lower sales of pendants compared to earrings and rings, and how can we address them?

Overall Analysis:



Insights from the Line and Stacked column Chart (Category Alias and sales by female):

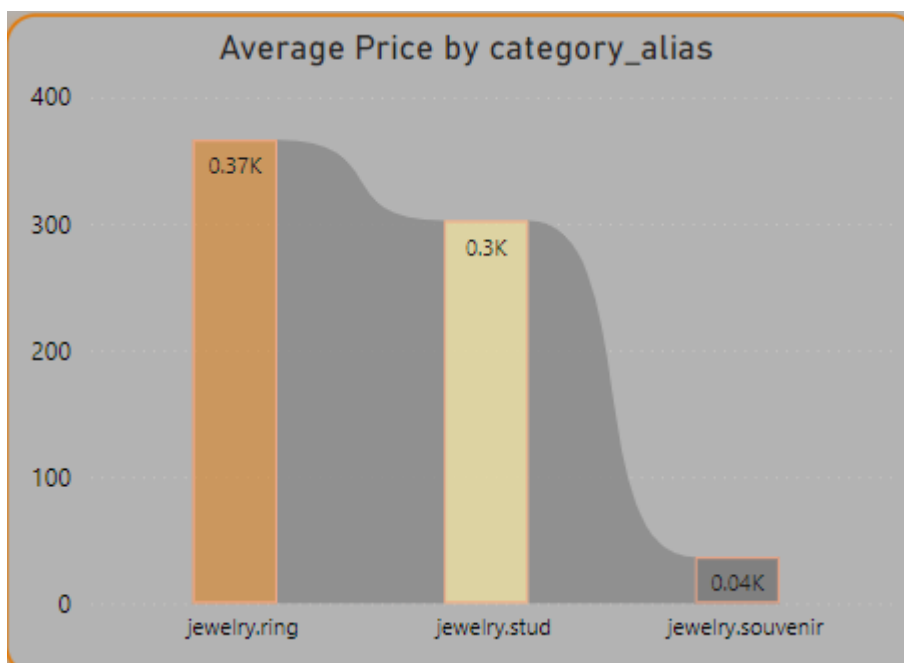
- Earrings lead in sales with **7.9M**, showing the highest demand among female customers.
- Rings follow with **5.5M**, indicating moderate demand.
- Pendants have the lowest sales at **1.4M**, suggesting an area for potential growth.

Questions:

Why are earrings significantly more popular than rings and pendants?

What strategies can be implemented to boost pendant sales?

How do customer preferences differ between these categories, and how can this knowledge be used to improve overall sales?



Insights from the Ribbon Chart (Category Alias and Average Price):

Rings have the highest average price at **0.37k**, indicating a premium positioning compared to other categories.

Studs follow with an average price of **0.3k**, suggesting moderate pricing.

Souvenirs have the lowest average price at **0.04k**, indicating they are likely budget-friendly or lower-cost items.

APPENDIX: BI DESIGN

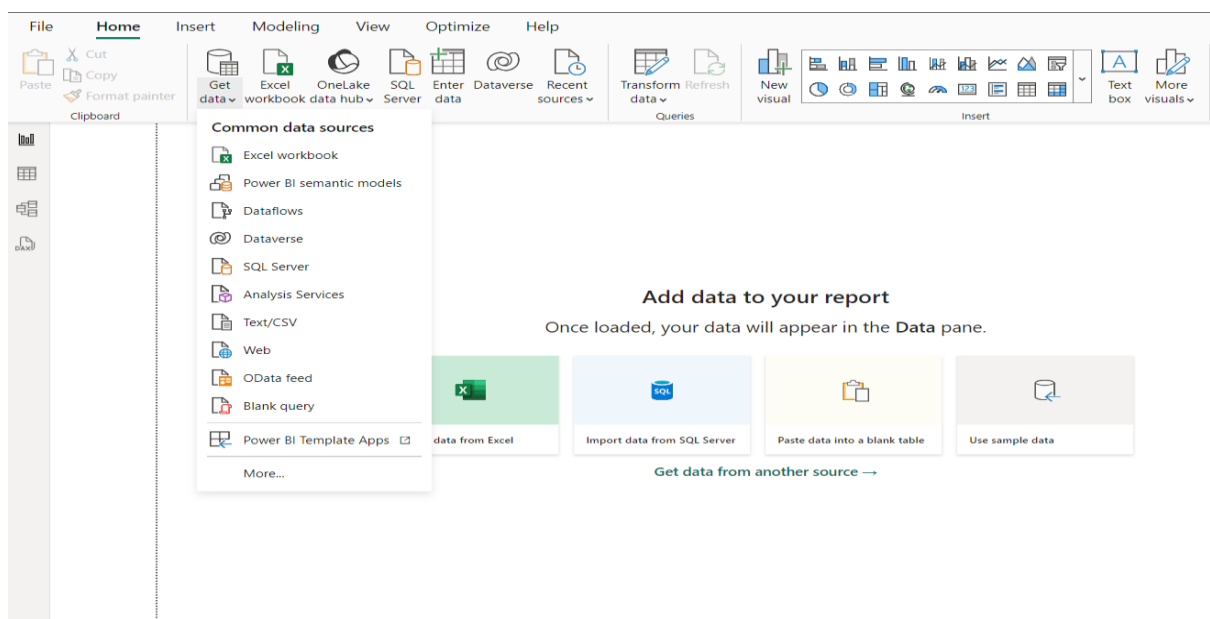
I. DATA PRE-PROCESSING AND DATA CLEANING

Load and Clean Dataset

Connecting to Power BI

Using Power BI Desktop, I can create dynamic reports and dashboards that show the most up-to-date data.

Import The Data:



Transform the data for clean it.

File Home Insert Modeling View Optimize Help

Get data Excel OneLake workbook data hub

Clipboard

jewelry.csv

File Origin: 1252: Western European (Windows) Delimiter: Comma Data Type Detection: Based on first 200 rows

date	order_id	product_id	quantity	category_id	category_alias	brand_id	price	user_id	gender
2018-12-01 11:40:29 UTC	1.92472E+18	1.8422E+18	1	1.80683E+18	jewelry.earring	0	561.51	1.51592E+18	
2018-12-01 17:38:31 UTC	1.9249E+18	1.80683E+18	1	1.80683E+18		null	212.14	1.51592E+18	
2018-12-02 13:53:42 UTC	1.92551E+18	1.84221E+18	1	1.80683E+18	jewelry.pendant	1	54.66	1.51592E+18	f
2018-12-02 17:44:02 UTC	1.92563E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant	0	88.9	1.51592E+18	f
2018-12-02 21:30:19 UTC	1.92574E+18	1.87394E+18	1	1.80683E+18	jewelry.necklace	0	417.67	1.51592E+18	
2018-12-02 22:09:34 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring	1	102.27	1.51592E+18	
2018-12-02 22:09:34 UTC	1.92576E+18	1.60002E+18	1	1.80683E+18	jewelry.earring	0	150.55	1.51592E+18	
2018-12-02 22:16:20 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring	0	83.42	1.51592E+18	
2018-12-03 07:03:49 UTC	1.92603E+18	1.8118E+18	1	1.80683E+18	jewelry.pendant	2	60.27	1.51592E+18	f
2018-12-03 09:48:34 UTC	1.92611E+18	1.83628E+18	1	1.80683E+18	jewelry.earring	1	217.4	1.51592E+18	
2018-12-03 09:48:34 UTC	1.92611E+18	1.87764E+18	1	1.80683E+18	jewelry.ring	1	180.71	1.51592E+18	
2018-12-03 09:48:34 UTC	1.92611E+18	1.87834E+18	1	1.80683E+18	jewelry.brooch	1	232.1	1.51592E+18	
2018-12-03 11:14:25 UTC	1.92616E+18	1.83955E+18	1	1.80683E+18	jewelry.ring	1	203.92	1.51592E+18	
2018-12-03 18:43:00 UTC	1.92638E+18	1.80746E+18	1	1.80683E+18	jewelry.earring	0	84.45	1.51592E+18	
2018-12-04 06:04:18 UTC	1.92672E+18	1.83975E+18	1	1.80683E+18	jewelry.ring	0	254.45	1.51592E+18	
2018-12-04 08:06:10 UTC	1.92679E+18	1.83809E+18	1	1.80683E+18	jewelry.earring	0	251.15	1.51592E+18	
2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1	154.88	1.51592E+18	
2018-12-04 12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1	154.88	1.51592E+18	
2018-12-04 17:18:19 UTC	1.92706E+18	1.83683E+18	1	1.80683E+18	jewelry.earring	0	217.11	1.51592E+18	
2018-12-04 19:55:22 UTC	1.92714E+18	1.80757E+18	1	1.80683E+18	jewelry.earring	1	102.6	1.51592E+18	f

Extract Table Using Examples Load Transform Data Cancel

Transformed the data

File Home Transform Add Column View Tools Help

Close & Apply New Source Recent Enter Data Data source settings Manage Parameters Refresh Preview Properties Advanced Editor Choose Columns Remove Columns Keep Rows Remove Rows Sort Split Column Group By Data Type: Text Use First Row as Headers Merge Queries Append Queries Combine Files Combine Text Analytics Vision Azure Mach

Queries [1] Table.TransformColumnTypes("#Promoted Headers",{{"date", type text}, {"order_id", Int64.Type}, {"product_id", Int64.Type}, {"quant

jewelry

date	order_id	product_id	quantity	category_id	category_alias
Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 99% Error 0% Empty < 1%	Valid 94% Error 0% Empty 6%
789 distinct, 645 unique	789 distinct, 645 unique	522 distinct, 295 unique	1 distinct, 0 unique	11 distinct, 0 unique	11 distinct, 2 unique
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
21 2018-12-04 20:10:22 UTC	1.92715E+18	1.80683E+18	1	1.80683E+18	jewelry.pendant

Data Processing:

Addressed missing and blank values in the dataset.

Replaced null values with appropriate replacements to ensure consistency and accuracy etc.

Convert appropriate columns to numeric or datetime formats.

The below date column time stamp format.

date

2018-12-01 11:40:29 UTC

- We need to separate the date and time columns for better analysis of trends and patterns.

Table.TransformColumnTypes(#"Promoted Headers",{{"date", type text}, {"order_id", Int64.Type}, {"product_id", Int64.Type}, {"quantity", Int64.Type}, {"category_id", Int64.Type}, {"category_alias", type text}, {"brand_id", Int64.Type}})

	date	order_id	product_id	quantity	category_id	category_alias	brand_id
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	

Table.TransformColumnTypes(#"Promoted Headers",{{"date", type text}, {"order_id", Int64.Type}, {"product_id", Int64.Type}, {"quantity", Int64.Type}, {"category_id", Int64.Type}, {"category_alias", type text}, {"brand_id", Int64.Type}})

	date	order_id	product_id	quantity	category_id	category_alias	brand_id
1	2018-12-01 11		1.8422E+18	1	1.80683E+18	jewelry.earring	
2	2018-12-01 17		1.80683E+18	1	1.80683E+18		
3	2018-12-02 13		1.84221E+18	1	1.80683E+18	jewelry.pendant	
4	2018-12-02 17		1.83557E+18	1	1.80683E+18	jewelry.pendant	
5	2018-12-02 21		1.87394E+18	1	1.80683E+18	jewelry.necklace	
6	2018-12-02 22			1	1.80683E+18	jewelry.earring	
7	2018-12-02 22			1	1.80683E+18	jewelry.earring	
8	2018-12-02 22			1	1.80683E+18	jewelry.earring	
9	2018-12-03 07			1	1.80683E+18	jewelry.pendant	
10	2018-12-03 09			1	1.80683E+18	jewelry.earring	
11	2018-12-03 09			1	1.80683E+18	jewelry.ring	
12	2018-12-03 09			1	1.80683E+18	jewelry.brooch	
13	2018-12-03 11		1.83955E+18	1	1.80683E+18	jewelry.ring	
14	2018-12-03 18		1.80746E+18	1	1.80683E+18	jewelry.earring	
15	2018-12-04 06		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	

- Replace with space

×

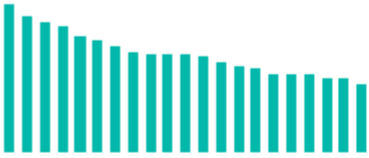


Select or enter delimiter

- Each occurrence of the delimiter

Insert special character ▾

Cancel

= TransformColumnTypes("#Split Column by Delimiter",{{"date.1", type date}, {"date.2", type time}, {"date.3", type text}})									
date.1	date.2	date.3	order_id	product_id	quantity	category_id			
<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>68 distinct, 8 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>786 distinct, 639 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>1 distinct, 0 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>789 distinct, 645 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>522 distinct, 295 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>1 distinct, 0 unique</div>	<div><div>Valid 100%</div><div>Error 0%</div><div>Empty 0%</div></div> <div></div> <div>11 distinct, 0 unique</div>			
1	01-12-2018	11:40:29 UTC	1.92472E+18	1.8422E+18	1				
2	01-12-2018	17:38:31	1.9249E+18	1.80683E+18	1				
3	02-12-2018	13:53:42 UTC	1.92551E+18	1.84221E+18	1				
4	02-12-2018	17:44:02 UTC	1.92563E+18	1.83557E+18	1				
5	02-12-2018	21:30:19 UTC	1.92574E+18	1.87394E+18	1				
6	02-12-2018	22:09:34	1.92576E+18	1.83557E+18	1				
7	02-12-2018	22:09:34 UTC	1.92576E+18	1.60002E+18	1				
8	02-12-2018	22:16:20 UTC	1.92576E+18	1.83557E+18	1				
9	03-12-2018	07:03:49 UTC	1.92603E+18	1.81118E+18	1				
10	03-12-2018	09:48:34	1.92611E+18	1.83628E+18	1				
11	03-12-2018	09:48:34 UTC	1.92611E+18	1.87764E+18	1				
12	03-12-2018	09:48:34 UTC	1.92611E+18	1.87834E+18	1				
13	03-12-2018	11:14:25 UTC	1.92616E+18	1.83955E+18	1				
14	03-12-2018	18:43:00 UTC	1.92638E+18	1.80746E+18	1				
15	04-12-2018	06:04:18 UTC	1.92672E+18	1.83975E+18	1				
16	04-12-2018	08:06:10 UTC	1.92679E+18	1.83809E+18	1				
17	04-12-2018	12:40:48 UTC	1.92692E+18	1.8744E+18	1				
18	04-12-2018	12:40:48 UTC	1.92692E+18	1.8744E+18	1				

= Table.TransformColumnTypes("#Split Column by Del			
	date.1	date.2	date.3
	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>68 distinct, 8 unique</p>	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>786 distinct, 639 unique</p>	<ul style="list-style-type: none"> Valid Error Empty  <p>1 distinct, 0 unique</p>
1	01-12-2018	11:40:29	UTC
2	01-12-2018	17:38:31	UTC
3	02-12-2018	13:53:42	UTC
4	02-12-2018	17:44:02	UTC
5	02-12-2018	21:30:19	UTC
6	02-12-2018	22:09:34	UTC
7	02-12-2018	22:09:34	UTC
8	02-12-2018	22:16:20	UTC
9	03-12-2018	07:03:49	UTC
10	03-12-2018	09:48:34	UTC
11	03-12-2018	09:48:34	UTC
12	03-12-2018	09:48:34	UTC
13	03-12-2018	11:14:25	UTC
14	03-12-2018	18:43:00	UTC
15	04-12-2018	06:04:18	UTC
16	04-12-2018	08:06:10	UTC
17	04-12-2018	12:40:48	UTC
18	04-12-2018	12:40:48	UTC
19	04-12-2018	17:18:19	UTC
20	04-12-2018	19:55:22	UTC
21	04-12-2018	20:10:22	UTC

fx = Table.ReplaceValue("#Replaced Value10", "", "0", Replacer.ReplaceValue, {"stone"})

	date.3	order_id	product_id	quantity	category_id	category_alias
1	11:40:29 UTC	1.92472E+18	1.8422E+18	1	1.80683E+18	jewelry.earring
2	17:38:31 UTC	1.9249E+18	1.80683E+18	1	1.80683E+18	0
3	13:53:42 UTC	1.92551E+18	1.84221E+18	1	1.80683E+18	jewelry.pendant
4	17:44:02 UTC	1.92563E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant
5	21:30:19 UTC	1.92574E+18	1.87394E+18	1	1.80683E+18	jewelry.necklace
6	22:09:34 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring
7	22:09:34 UTC	1.92576E+18	1.60002E+18	1	1.80683E+18	jewelry.earring
8	22:16:20 UTC	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant
9	07:03:49 UTC	1.92603E+18	1.8118E+18	1	1.80683E+18	jewelry.pendant
10	09:48:34 UTC	1.92611E+18	1.83628E+18	1	1.80683E+18	jewelry.earring
11	09:48:34 UTC	1.92611E+18	1.87764E+18	1	1.80683E+18	jewelry.ring
12	09:48:34 UTC	1.92611E+18	1.87834E+18	1	1.80683E+18	jewelry.brooch
13	11:14:25 UTC	1.92616E+18	1.83955E+18	1	1.80683E+18	jewelry.ring
14	18:43:00 UTC	1.92638E+18	1.80746E+18	1	1.80683E+18	jewelry.earring
15	06:04:18 UTC	1.92672E+18	1.83975E+18	1	1.80683E+18	jewelry.ring
16	08:06:10 UTC	1.92679E+18	1.83809E+18	1	1.80683E+18	jewelry.earring
17	12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring
18	12:40:48 UTC	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring
19	17:18:19 UTC	1.92706E+18	1.83683E+18	1	1.80683E+18	jewelry.earring
20	19:55:22 UTC	1.92714E+18	1.80757E+18	1	1.80683E+18	jewelry.earring
21	20:10:22 UTC	1.92715E+18	1.80683E+18	1	1.80683E+18	jewelry.pendant

*Change the data type to text

*The **Category** column contains null values, which should be replaced with 0.

fx = Table.TransformColumnTypes("#Split Column by Delimiter",{"date.1", type date}, {"date.2", type time}, {"date.3", type text})

	order_id	product_id	quantity	category_id	category_alias	brand_id
1	1.92472E+18	1.8422E+18	1	1.80683E+18	jewelry.earring	0
2	1.9249E+18	1.80683E+18	1	1.80683E+18	jewelry.pendant	1
3	1.92551E+18	1.84221E+18	1	1.80683E+18	jewelry.pendant	0
4	1.92563E+18	1.83557E+18	1	1.80683E+18	jewelry.necklace	0
5	1.92574E+18	1.87394E+18	1	1.80683E+18	jewelry.earring	1
6	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.earring	0
7	1.92576E+18	1.60002E+18	1	1.80683E+18	jewelry.earring	0
8	1.92576E+18	1.83557E+18	1	1.80683E+18	jewelry.pendant	2
9	1.92603E+18	1.8118E+18	1	1.80683E+18	jewelry.earring	1
10	1.92611E+18	1.83628E+18	1	1.80683E+18	jewelry.ring	1
11	1.92611E+18	1.87764E+18	1	1.80683E+18	jewelry.brooch	1
12	1.92611E+18	1.87834E+18	1	1.80683E+18	jewelry.ring	1
13	1.92616E+18	1.83955E+18	1	1.80683E+18	jewelry.earring	0
14	1.92638E+18	1.80746E+18	1	1.80683E+18	jewelry.ring	0
15	1.92672E+18	1.83975E+18	1	1.80683E+18	jewelry.earring	0
16	1.92679E+18	1.83809E+18	1	1.80683E+18	jewelry.earring	1
17	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1
18	1.92692E+18	1.8744E+18	1	1.80683E+18	jewelry.ring	1
19	1.92706E+18	1.83683E+18	1	1.80683E+18	jewelry.earring	0
20	1.92714E+18	1.80757E+18	1	1.80683E+18	jewelry.earring	1
21	1.92715E+18	1.80683E+18	1	1.80683E+18	jewelry.pendant	1

×

Replace Values

Replace one value with another in the selected columns.

Value To Find

Replace With

OK

Cancel

*The **Category Alias** contains Blank values, which should be replaced with 0.

×

✓

fx

= Table.ReplaceValue("#"Changed Type1",null,0,Replacer.ReplaceValue,{"category_id"})

order_id	product_id	quantity	category_id	category_alias	brand_id	price
Valid 100%	Valid 100%	Valid 100%	Sort Ascending	Valid 90%	Valid 100%	Valid 100%
Error 0%	Error 0%	Error 0%	Sort Descending	Error 0%	Error 0%	Error 0%
Empty 0%	Empty 0%	Empty 0%	Clear Sort	Empty 10%	Empty 0%	Empty 0%
39 distinct, 645 unique	522 distinct, 295 unique	1 distinct, 0 unique	Clear Filter	10 distinct, 2 unique	511 distinct, 320 unique	
1	1.92472E+18	1.8422E+18	Remove Empty			
2	1.9249E+18	1.80683E+18	Text Filters			
3	1.92551E+18	1.84221E+18	Search			
4	1.92563E+18	1.83557E+18	(Select All)			
5	1.92574E+18	1.87394E+18	(blank)			
6	1.92576E+18	1.83557E+18	202.36			
7	1.92576E+18	1.60002E+18	230.12			
8	1.92576E+18	1.83557E+18	electronics.clocks			
9	1.92603E+18	1.8118E+18	jewelry.bracelet			
10	1.92611E+18	1.83628E+18	jewelry.brooch			
11	1.92611E+18	1.87764E+18	jewelry.earring			
12	1.92611E+18	1.87834E+18	jewelry.necklace			
13	1.92616E+18	1.83955E+18	jewelry.pendant			
14	1.92638E+18	1.80746E+18	jewelry.ring			
15	1.92672E+18	1.83975E+18	jewelry.souvenir			
16	1.92679E+18	1.83809E+18	List may be incomplete. Load more			
17	1.92692E+18	1.8744E+18	OK			
18	1.92692E+18	1.8744E+18	Cancel			
19	1.92706E+18	1.83683E+18				
20	1.92714E+18	1.80757E+18				
21	1.92715E+18	1.80683E+18				

×

Replace Values

Replace one value with another in the selected columns.

Value To Find

Replace With

Advanced options

OK

Cancel

*The **brand_id** contains Null values, which should be replaced with 0.

Table.ReplaceValue("#Replaced Value","", "0",Replacer.ReplaceValue,{"category_alias"})

The screenshot shows a data table with columns: product_id, quantity, category_id, category_alias, and brand_id. A 'Replace Values' dialog box is open, allowing the user to replace 'null' with '0' in the selected columns. The dialog includes fields for 'Value To Find' (set to 'null') and 'Replace With' (set to '0'), along with 'OK' and 'Cancel' buttons.

Accordingly Replace the below

*The **price** contains Null values, which should be replaced with 0.

*The **user_id** contains Null values, which should be replaced with 0.

*The Gender_**id** contains Blank values, which should be replaced with 0.

*In gender column replace f with female and m with male.

The screenshot shows a 'Replace Values' dialog box. The 'Value To Find' field contains 'f' and the 'Replace With' field contains 'Female'. There is an 'Advanced options' link and 'OK' and 'Cancel' buttons.

*The color contains Blank values, which should be replaced with 0.

*The material contains Blank values, which should be replaced with 0.

*The stone contains Blank values, which should be replaced with 0.

date.1	date.2	order_id	product_id	quantity	category_id	category.alias	brand_id	price	user_id	gender	color	material	st
28 January 2019	12:18:39	19667539503430000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	148111609137463000				
18 February 2019	16:07:19	1982110782040570000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	151591562526119000				
18 February 2019	16:07:19	1982110782040570000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	151591562526119000				
22 June 2019	09:28:45	2071782360591890000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625255950000	0	red	gold	
28 June 2019	14:43:06	2076289239837830000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625248990000	0	red	gold	
12 July 2019	14:31:45	2086430387298370000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625311540000	0	red	gold	
28 July 2019	04:45:38	2097731790889280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625192540000	0	red	gold	
17 August 2019	16:37:45	2112585731942640000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625208570000	0	red	gold	
25 August 2019	09:54:46	2118181104597860000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625226850000	0	red	gold	
04 September 2019	18:48:13	2125697358124540000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625258540000	0	red	gold	
11 October 2019	12:43:57	2152330721060970000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625039970000	0	red	gold	
15 October 2019	11:52:35	215520396353470000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625218850000	0	red	gold	
16 November 2019	09:40:13	2178330169792520000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625217790000	0	red	gold	
20 November 2019	13:59:40	2181359859067250000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625323720000	0	red	gold	
21 November 2019	09:24:01	2181945896525100000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625277670000	0	red	gold	
21 November 2019	15:11:38	2182120858376470000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625175580000	0	red	gold	
25 November 2019	08:07:44	218480660291280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625201830000	0	red	gold	
25 November 2019	16:36:36	218480660291280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625201830000	0	red	gold	

date.1	date.2	order_id	product_id	quantity	category_id	category.alias	brand_id	price	user_id	gender	color	material	st
28 January 2019	12:18:39	19667539503430000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	148111609137463000				
18 February 2019	16:07:19	1982110782040570000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	151591562526119000				
18 February 2019	16:07:19	1982110782040570000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	151591562526119000				
22 June 2019	09:28:45	2071782360591890000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625255950000	0	red	gold	
28 June 2019	14:43:06	2076289239837830000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625248990000	0	red	gold	
12 July 2019	14:31:45	2086430387298370000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625311540000	0	red	gold	
28 July 2019	04:45:38	2097731790889280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625192540000	0	red	gold	
17 August 2019	16:37:45	2112585731942640000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625208570000	0	red	gold	
25 August 2019	09:54:46	2118181104597860000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625226850000	0	red	gold	
04 September 2019	18:48:13	2125697358124540000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625258540000	0	red	gold	
11 October 2019	12:43:57	2152330721060970000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625039970000	0	red	gold	
15 October 2019	11:52:35	215520396353470000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625218850000	0	red	gold	
16 November 2019	09:40:13	2178330169792520000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625217790000	0	red	gold	
20 November 2019	13:59:40	2181359859067250000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625323720000	0	red	gold	
21 November 2019	09:24:01	2181945896525100000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625277670000	0	red	gold	
21 November 2019	15:11:38	2182120858376470000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625175580000	0	red	gold	
25 November 2019	08:07:44	218480660291280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625201830000	0	red	gold	
25 November 2019	16:36:36	218480660291280000	1956663847725040000	7	1806829201882340000	jewelry.ring	0	273.84	1515915625201830000	0	red	gold	

Creating Dim tables

Dim Date



Choose Columns

Choose the columns to keep



☒ (Select All Columns)
 ☒ date

Dim_Color table

TABLE.RenameColumns(# Added Index ,)

	A ^B _C Color	1 ² ₃ Color_ID
	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>6 distinct, 6 unique</p>	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>6 distinct, 6 unique</p>
1	red	1
2	yellow	2
3	white	3
4	0	4
5	unknown-color	5
6	black	6

Creating Material Dim table

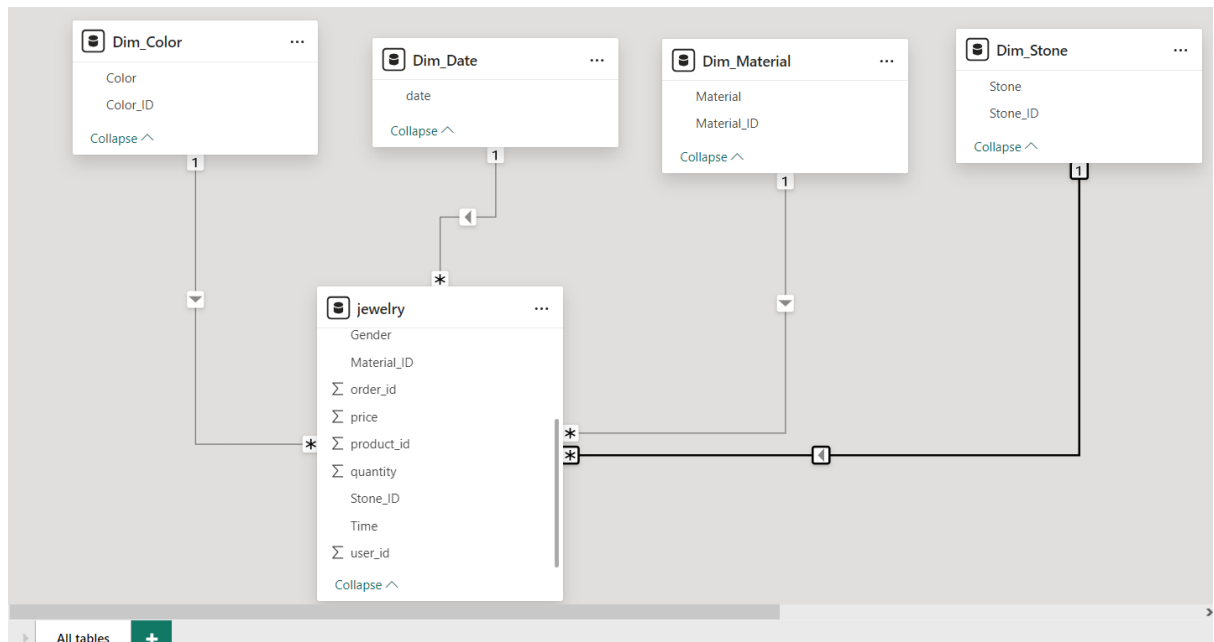
	A ^B _C Material	1 ² ₃ Material_ID
	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>4 distinct, 4 unique</p>	<ul style="list-style-type: none"> Valid 100% Error 0% Empty 0%  <p>4 distinct, 4 unique</p>
1	gold	50
2	silver	51
3	0	52
4	platinum	53

Creating stone dim table

	A ^B _C Stone	1 ² ₃ Stone_ID
	<div> <div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div> </div> <div>31 distinct, 31 unique</div>	<div> <div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div> </div> <div>31 distinct, 31 unique</div>
1	diamond	200
2	0	202
3	sapphire	204
4	amethyst	206
5	fianit	208
6	pearl	210
7	quartz	212
8	topaz	214
9	garnet	216
10	quartz_smoky	218
11	ruby	220

Data Modelling:

- Since we have only one table, there is no need to build additional relationships.



Create Measures and Calculated Columns

This measure calculates the total quantity of products sold.

```
1 Total Sales = SUMX(jewelry,jewelry[quantity]*jewelry[price])

. Total Quantity Sold = SUM(jewelry[quantity])
```

This measure calculates the average price of items sold.

```
Average Price = AVERAGE(jewelry[price])
```

This measure counts the number of distinct orders.

```
Total Orders = DISTINCTCOUNT(jewelry[order_id])
```

This measure calculates the total sales for each category in the category alias column.

```
Sales by Category =
CALCULATE( [Total Sales], ALLEXCEPT(jewelry, jewelry[category_alias]))
```

This measure calculates the total sales for each brand in the brand_id column.

```
Sales by Brand = CALCULATE([Total Sales], ALLEXCEPT(jewelry, jewelry[brand_id]))
```


This measure calculates the total sales for a particular gender.

```
Sales by Female = CALCULATE([Total Sales], jewelry[gender] = "Female")
```

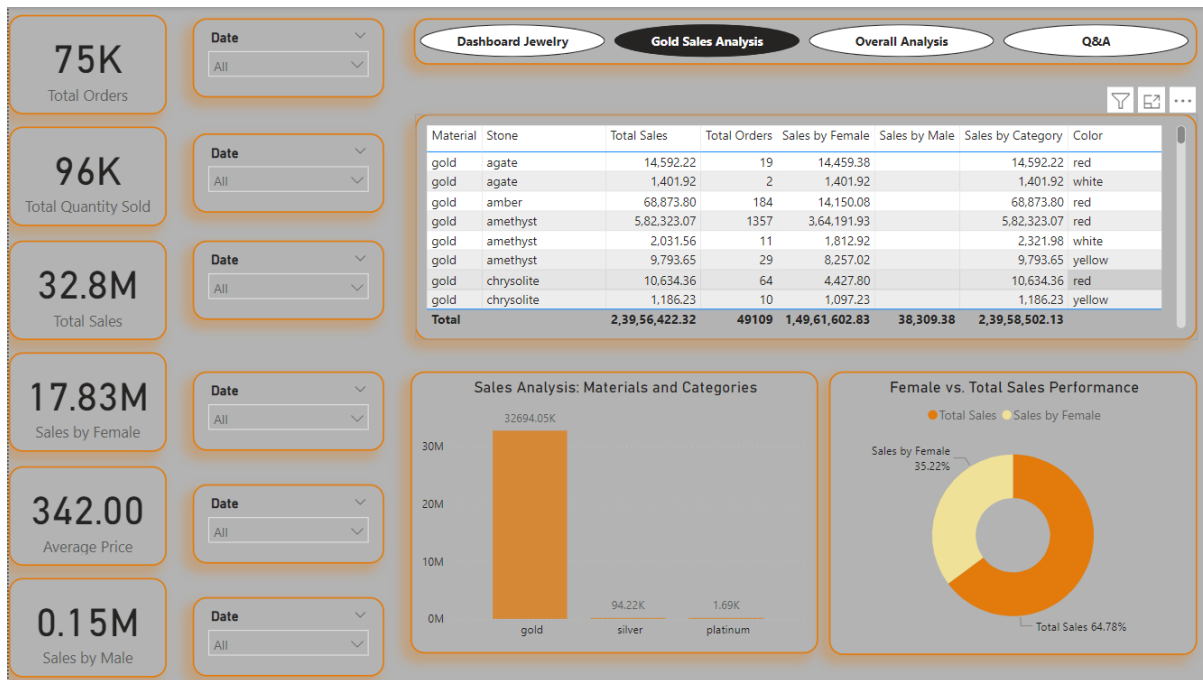
```
Sales by Male = CALCULATE([Total Sales], jewelry[gender] = "Male")
```

Dashboard Visualizations:

Home page.



Gold Sales Analysis:



Customer Analysis:

Insights from the Line and Stacked column Chart (Category Alias and sales by female):

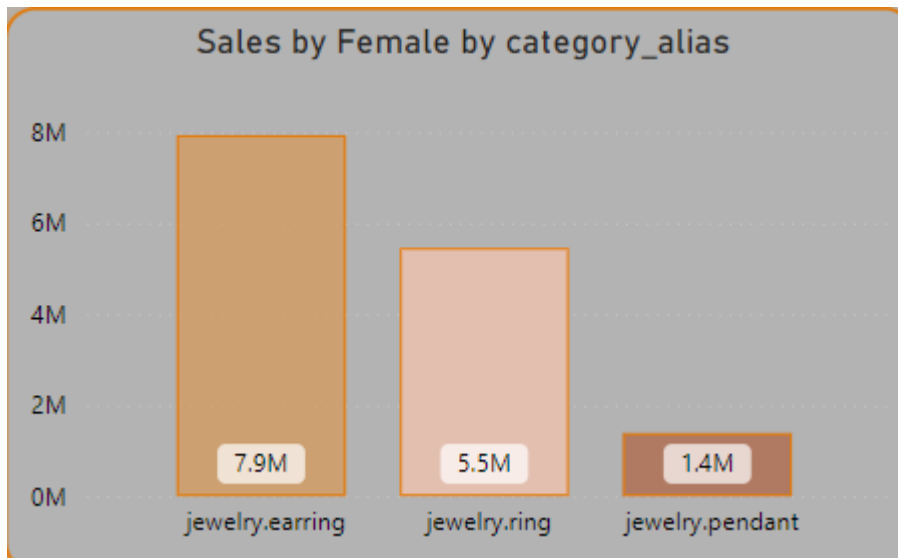
- Earrings lead with 7.9M in sales, showing strong performance among female customers.
- Rings follow with 5.5M in sales, indicating a significant contribution.
- Pendants have a smaller share, with 1.4M in sales, suggesting lower demand from female customers for this category.

Questions:

Why do earrings have the highest sales among female customers, and how can sales of pendants be improved?

What factors might be causing the lower sales of pendants compared to earrings and rings, and how can we address them?

Overall Analysis:



Insights from the Line and Stacked column Chart (Category Alias and sales by female):

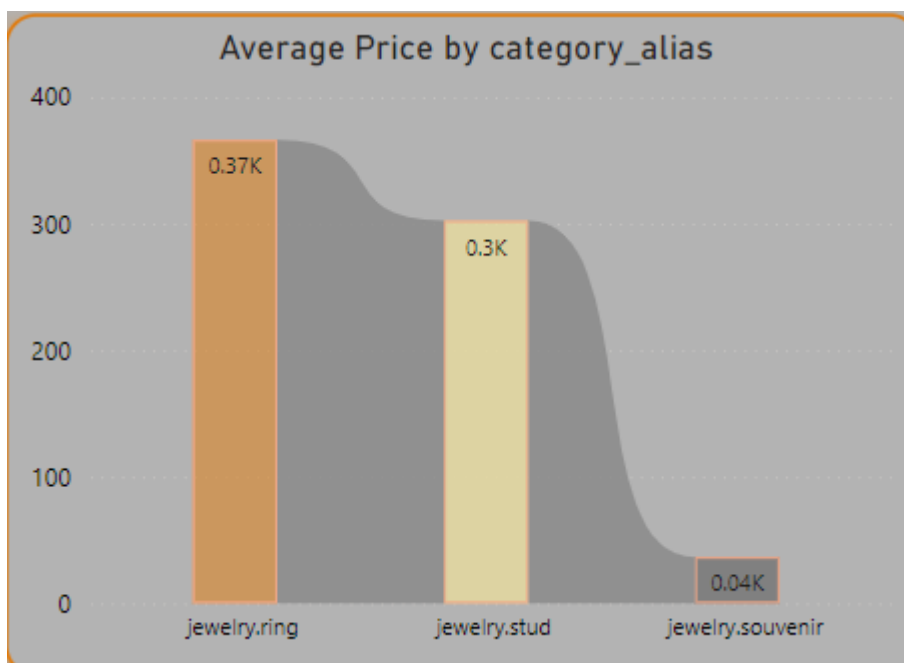
- Earrings lead in sales with **7.9M**, showing the highest demand among female customers.
- Rings follow with **5.5M**, indicating moderate demand.
- Pendants have the lowest sales at **1.4M**, suggesting an area for potential growth.

Questions:

Why are earrings significantly more popular than rings and pendants?

What strategies can be implemented to boost pendant sales?

How do customer preferences differ between these categories, and how can this knowledge be used to improve overall sales?



Insights from the Ribbon Chart (Category Alias and Average Price):

Rings have the highest average price at **0.37k**, indicating a premium positioning compared to other categories.

Studs follow with an average price of **0.3k**, suggesting moderate pricing.

Souvenirs have the lowest average price at **0.04k**, indicating they are likely budget-friendly or lower-cost items.

Questions:

1. Why do rings have a higher average price compared to studs and souvenirs?
2. How can the pricing strategy for souvenirs be adjusted to improve their perceived value?
3. Are these price differences impacting customer preferences and overall sales distribution?

