



Project Abstract

Welcome to my immersive SQL-driven *Paytm Mall Epurchase Data Analysis project* done as part of my Data Analyst Internship with PSYLIQ, where the fusion of data science and healthcare opens doors to a realm of possibilities! This program is a gateway to unlocking the potential of SQL in dissecting and comprehending healthcare data. As an MSc grad in Physics passionate about software development and Data Analytics, this internship promises an invaluable opportunity to wield my skills and broaden my horizons. Throughout this enriching journey, we embark on an exploration of a multifaceted dataset encompassing pivotal elements such as Paytm Mall Epurchase (S#no, Name, Shipping_city, Category_Grouped, Category, Sub_category, Product_Gender, Segment, Class, Family, Brand, Brick, Item_NM, Color, Size, Sale_Flag, Payment_Method, coupon_money_effective, Coupon_Percentage, Quantity, Cost_Price, Item_Price, Special_Price_effective, paid_pr_effective, Value_CM1, Value_CM2, Special_price, Paid_pr). This comprehensive real-world dataset mirrors the intricacies of healthcare data, serving as a robust platform to sharpen your SQL prowess.

Paytm Mall Epurchase Data Assessment Details

1. What does the "Category_Grouped" column represent, and how many unique categories are there?

```
SELECT COLUMN_NAME
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME = 'purchase';
```

2. Can you list the top 5 shipping cities in terms of the number of orders?

```
SELECT TOP 5 Shipping_city, COUNT(*) AS OrdersCount
FROM Purchase
GROUP BY Shipping_city
ORDER BY OrdersCount DESC;
```

3. Show me a table with all the data for products that belong to the "Electronics" category.

```
SELECT *
FROM Purchase
WHERE Category_Grouped = 'Electronics';
```

4. Filter the data to show only rows with a "Sale_Flag" of 'Yes'.

```
SELECT *
FROM Purchase
WHERE Category_Grouped = 'Electronics' AND Sale_Flag = 'Yes';
```

5. Sort the data by "Item_Price" in descending order. What is the most expensive item?

```
SELECT TOP 1 *
FROM Purchase
ORDER BY Item_Price DESC;
```

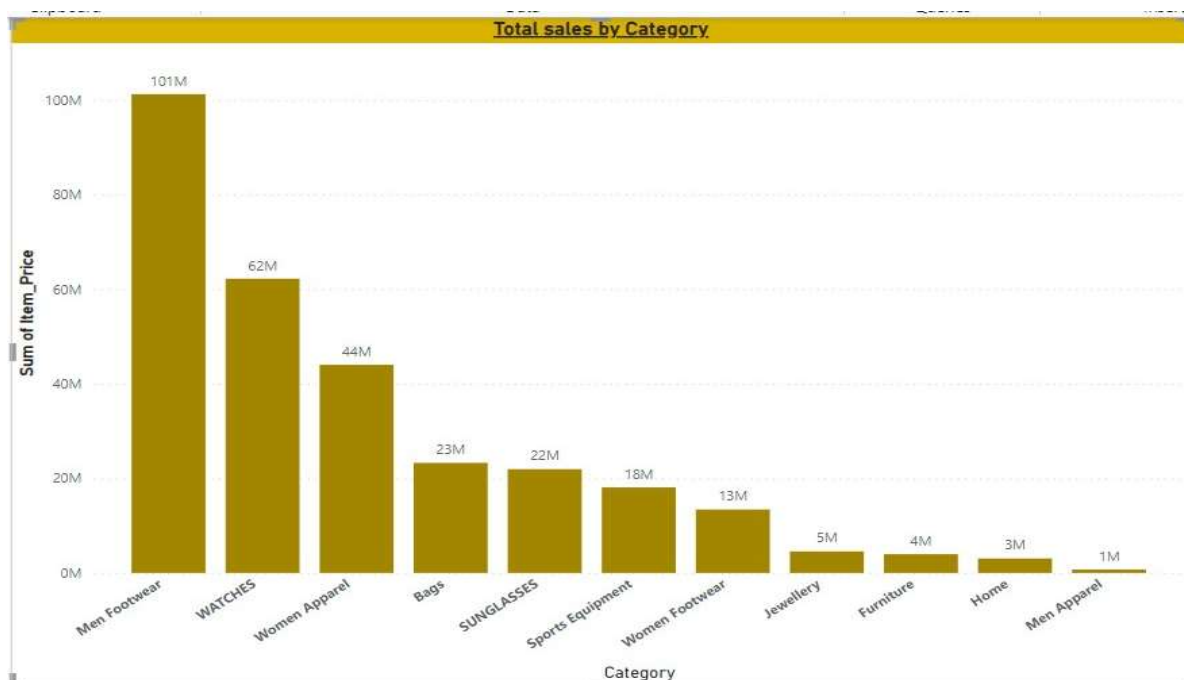
6. Apply conditional formatting to highlight all products with a "Special_Price_effective" value below \$50 in red.

```
SELECT TOP 1 *  
FROM Purchase  
ORDER BY Item_Price DESC;  
  
SELECT *,  
    CASE  
        WHEN Special_Price_effective < 50 THEN 'Below $50'  
        ELSE 'Above $50'  
    END AS Price_Category  
FROM Purchase;
```

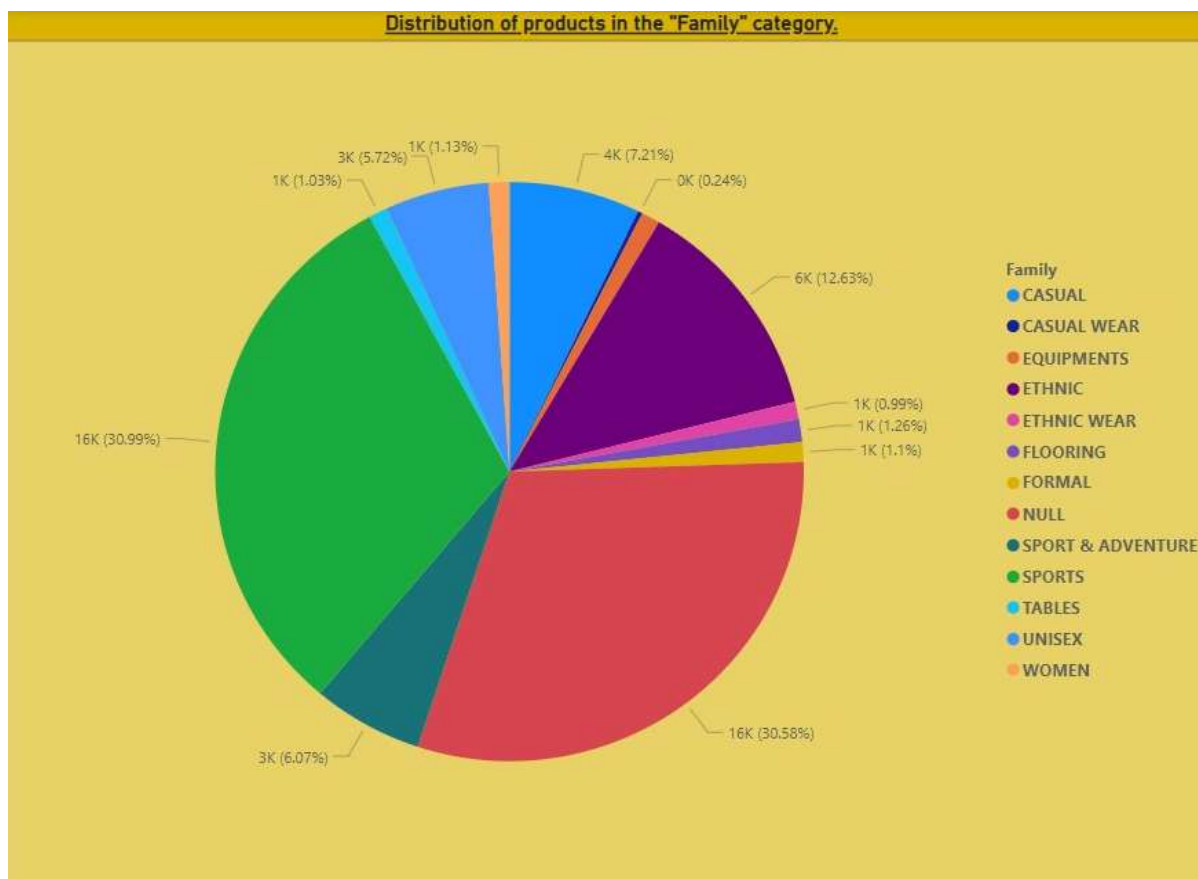
7. Create a pivot table to find the total sales value for each category.

```
SELECT Category_Grouped, SUM(Item_Price) AS TotalSalesValue  
FROM Purchase  
GROUP BY Category_Grouped;
```

8. Create a bar chart to visualize the total sales for each category.



9. Create a pie chart to show the distribution of products in the "Family" category.



10. Ensure that the "Payment_Method" column only contains valid payment methods (e.g., Visa, MasterCard).

```
UPDATE Purchase
SET Payment_Method =
CASE
    WHEN Payment_Method NOT IN ('Visa', 'MasterCard') THEN NULL --
    ELSE Payment_Method
END;
```

Replace with appropriate value

11. Calculate the average "Quantity" sold for products in the "Clothing" category, grouped by "Product_Gender."

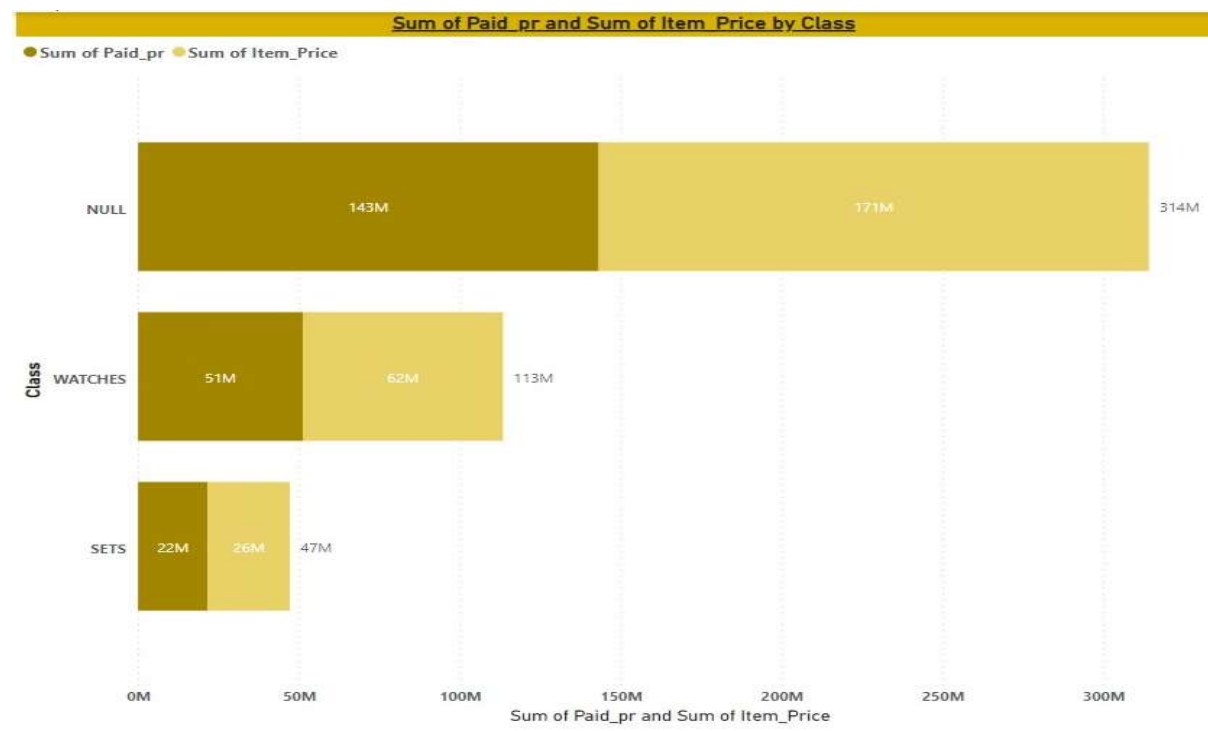
```
SELECT Product_Gender, AVG(Quantity) AS AverageQuantitySold
FROM Purchase
WHERE Category_Grouped = 'Clothing'
GROUP BY Product_Gender;
```

12. Find the top 5 products with the highest "Value_CM1" and "Value_CM2" ratios. Create a chart to visualize this data.

```
SELECT TOP 5 *,
       Value_CM1 / NULLIF(Value_CM2, 0) AS Ratio_Value_CM1_to_CM2
FROM Purchase
WHERE Value_CM2 <> 0
ORDER BY Ratio_Value_CM1_to_CM2 DESC;
```

13. Identify the top 3 "Class" categories with the highest total sales. Create a stacked bar chart to represent this data.

```
SELECT TOP 3 Class, SUM(Item_Price) AS TotalSales
FROM Purchase
GROUP BY Class
ORDER BY TotalSales DESC;
```



14. Use VLOOKUP or INDEX-MATCH to retrieve the "Color" of a product with a specific "Item_NM."

```
SELECT Color
FROM Purchase
WHERE Item_NM = 'your_specific_Item_NM';
```

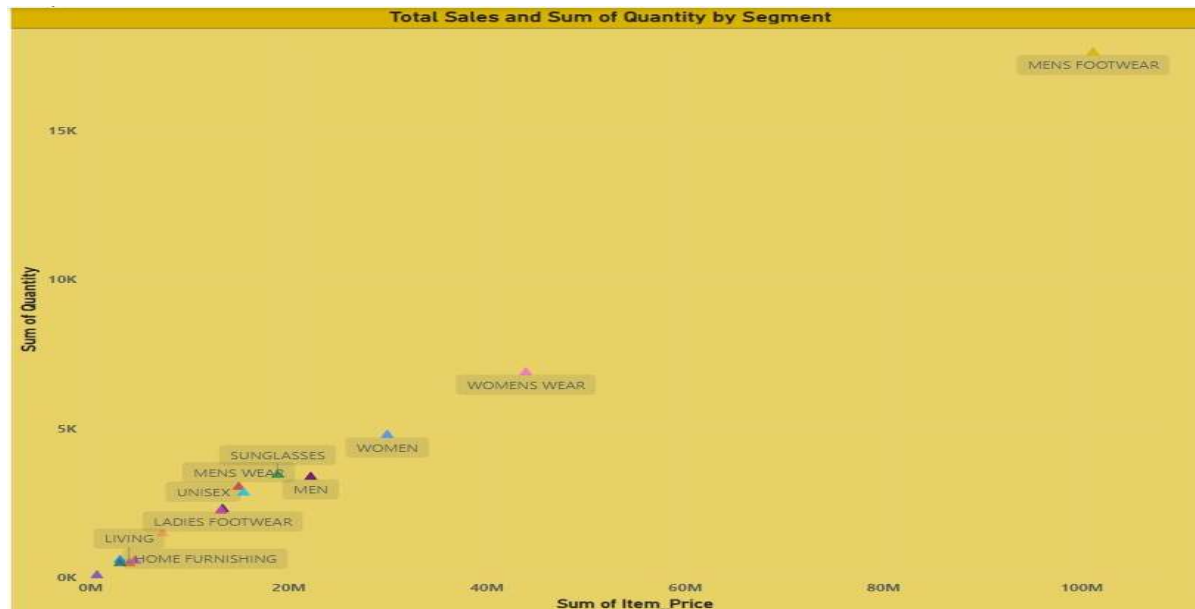
15. Calculate the total "coupon_money_effective" and "Coupon_Percentage" for products in the "Electronics" category.

```
SELECT
    SUM(coupon_money_effective) AS TotalCouponMoney,
    SUM(Coupon_Percentage) AS TotalCouponPercentage
FROM Purchase
WHERE Category_Grouped = 'Electronics';
```

16. Perform a time series analysis to identify the month with the highest total sales.

```
--SELECT top 1
--    DATEPART(month, YourDateColumn) AS SalesMonth,
--    SUM(Item_Price) AS TotalSales
--FROM Purchase
--GROUP BY DATEPART(month, YourDateColumn)
--ORDER BY TotalSales DESC
```

17. Calculate the total sales for each "Segment" and create a scatter plot to visualize the relationship between "Item_Price" and "Quantity" in this data.



18. Use the AVERAGEIFS function to find the average "Item_Price" for products that have a "Sale_Flag" of 'Yes.'

```
SELECT AVG(Item_Price) AS AverageItemPrice
FROM Purchase
WHERE Sale_Flag = 'Yes';
```

19. Identify products with a "Paid_pr" higher than the average in their respective "Family" and "Brand" groups.

```
SELECT *
FROM Purchase P1
WHERE Paid_pr > (
    SELECT AVG(Paid_pr)
    FROM Purchase P2
    WHERE P1.Family = P2.Family AND P1.Brand = P2.Brand
);
```

20. Create a pivot table to show the total sales for each "Color" within the "Clothing" category and use conditional formatting to highlight the highest sales.

```
SELECT  
    Color,  
    SUM(Item_Price) AS TotalSales  
FROM Purchase  
WHERE Category_Grouped = 'Clothing'  
GROUP BY Color;
```


RESULTS from SQL Server

Query executed successfully. CRYPTOGRAPHIC-A\SQLSERVER ... CRYPTOGRAPHIC-A\Crypto... master 00:00:03 74,453 rows

Query executed successfully. CRYPTOGRAPHIC-A\SQLSERVER ... CRYPTOGRAPHIC-A\Crypto... master 00:00:03 74,453 rows