# E-commerce SQL Analysis

We have been provided with the sales, product and customer data . Following are the common questions that the stakeholders will be interested in. It gives us some important KPIs to have a look upon

We have used SQL (Big Query) here to do the analysis.

The dataset can be found here :[Dataset](https://drive.google.com/drive/folders/1AXsgQUD8gMRp0zHIF88hizhWMfwJ9ivr?usp=sharing)

Data Description -[PR2-Data-Dictionary- SQL Analysis](https://docs.google.com/document/d/1V8IoRnJipw8EWmyWesc9CRWiSgODyC6vORRvAI0vAEg/edit?usp=sharing)

**I. Basic Metrics from the Data**

* **Total Orders, Number of Customers, Total Sales, Count of Stores**

**Query:**

SELECT COUNT(DISTINCT BASKET\_ID) as Total\_Orders

FROM ecommerce\_sql.transaction\_data;

SELECT COUNT(DISTINCT PRODUCT\_ID) as Total\_Products\_Sold

FROM ecommerce\_sql.transaction\_data;

SELECT COUNT(DISTINCT household\_key) as Total\_Customers

FROM ecommerce\_sql.transaction\_data;

SELECT COUNT(DISTINCT STORE\_ID) as Num\_of\_Stores

FROM ecommerce\_sql.transaction\_data;

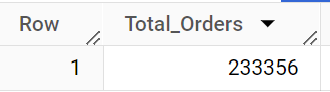
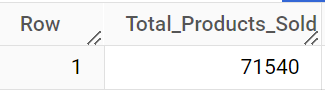
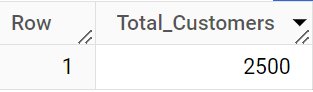
SELECT ROUND(SUM(SALES\_VALUE),2) as Total\_Sales\_Value

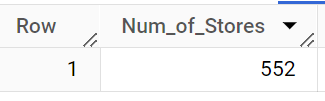
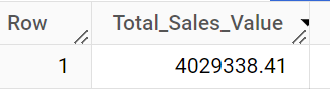
FROM ecommerce\_sql.transaction\_data;

SELECT SUM(QUANTITY)/COUNT (DISTINCT BASKET\_ID) as Average\_basket\_Size

FROM ecommerce\_sql.transaction\_data;

**Result:**

** ** ****

**II. Is there a growing trend in the no. of orders placed over the weeks?**

**Query:**

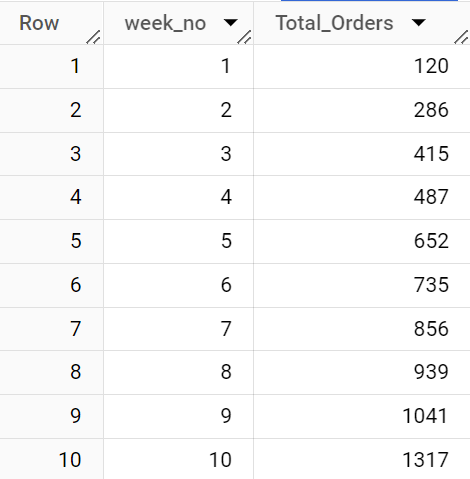
select week\_no,count(distinct basket\_id) as Total\_Orders

from ecommerce\_sql.transaction\_data

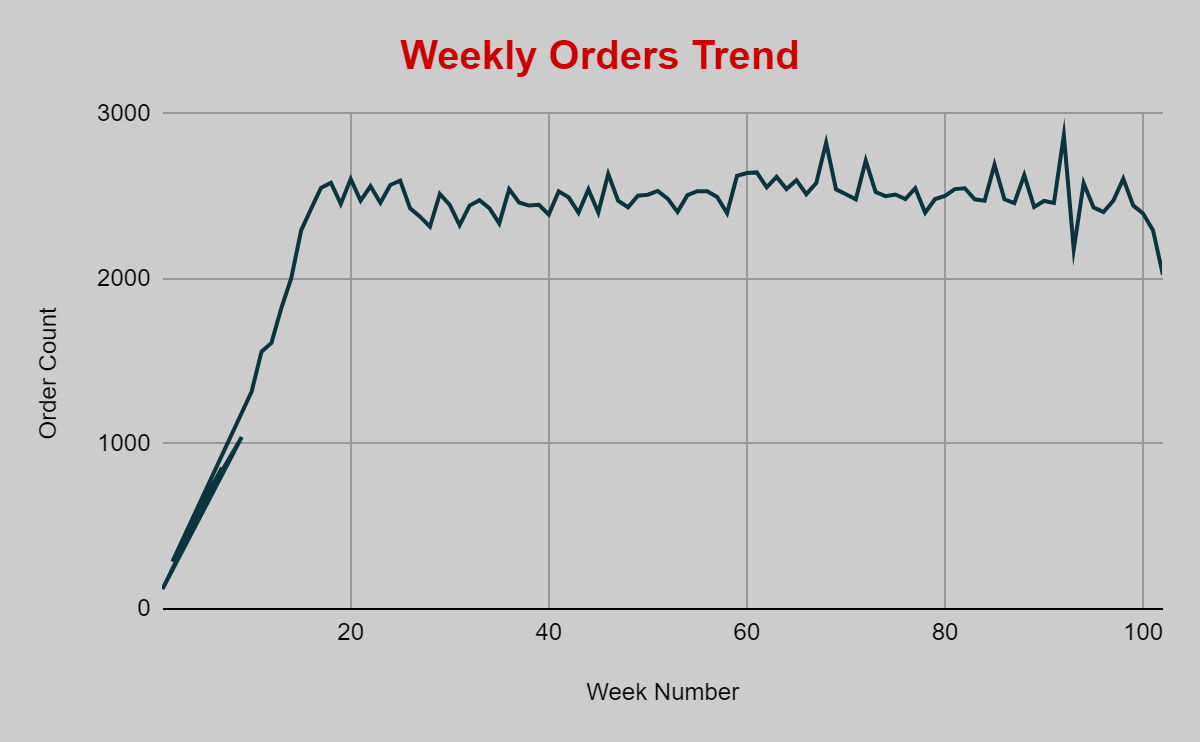
group by 1

order by 1;

**Result:**

****

**Insights:**



* There is a steady increase in orders till week 20.
* From there with few ups and downs the order trend has been almost same.

**Recommendations:**

* Offer seasonal discounts on the products specific to a particular season to boost the sales.
* Implement new marketing strategies to attract untapped customer base.

**B.** During what time of the day, do the customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

● 0-6 hrs: Dawn ● 7-12 hrs: Mornings ● 13-18 hrs: Afternoon ● 19-23 hrs: Night

**Query:**

select

  case

    when trans\_time between 0000 and 0600 then 'Dawn'

    when trans\_time between 0601 and 1200 then 'Morning'

    when trans\_time between 1201 and 1800 then 'Afternoon'

    when trans\_time between 1801 and 2359 then 'Night'

    end as Duration\_Of\_Day,

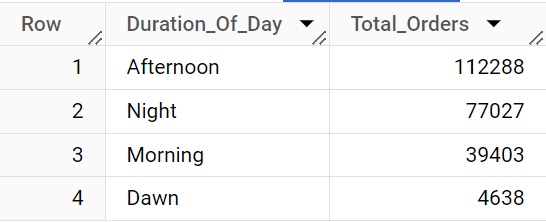
  count(distinct basket\_id) as Total\_Orders

from ecommerce\_sql.transaction\_data

group by 1

order by 2 desc;

**Result:**

****

**Insights:**

* Most of the orders from customers are placing the orders during Afternoon.
* During Morning & Night good number of orders being placed.
* Least number of orders are being placed during Dawn.

**Recommendations:** Since most of the orders are being placed during afternoon, business can run more discounted sales or promotional offers during this time to increase the number of sales.

**III. Top 5 Selling Products by Volume and Revenue**

**Query:**

SELECT PRODUCT\_ID, SUM(QUANTITY) AS Total\_Quantity\_Sold

FROM ecommerce\_sql.transaction\_data

GROUP BY PRODUCT\_ID

ORDER BY Total\_Quantity\_Sold DESC

LIMIT 5;

SELECT PRODUCT\_ID, ROUND(SUM(SALES\_VALUE),2) AS Total\_Revenue

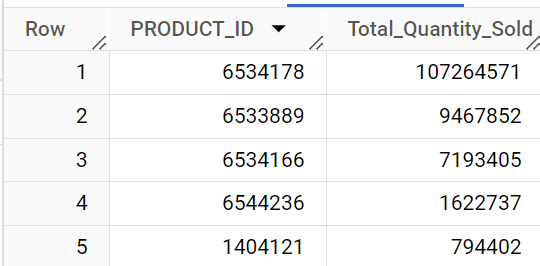
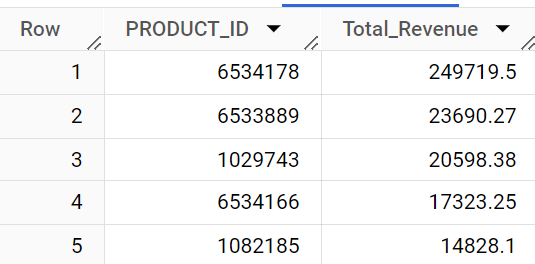
FROM ecommerce\_sql.transaction\_data

GROUP BY PRODUCT\_ID

ORDER BY Total\_Quantity\_Sold DESC

LIMIT 5;

**Result:**

**** ****

**IV. Top Stores by Revenue**

**Query:**

SELECT STORE\_ID,

    COUNT(DISTINCT BASKET\_ID) AS Total\_Orders,

    ROUND(SUM(SALES\_VALUE),2) AS Total\_Revenue,

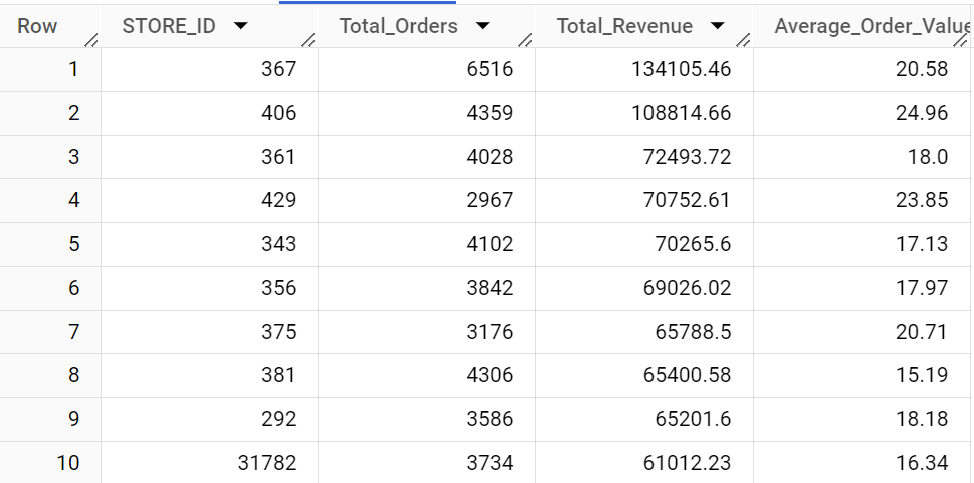
    ROUND(SUM(SALES\_VALUE)/COUNT(DISTINCT BASKET\_ID),2)AS Average\_Order\_Value,

FROM ecommerce\_sql.transaction\_data

GROUP BY STORE\_ID

ORDER BY Total\_Revenue Desc;

**Result:**

****

**V. Purchasing Behaviour based on Demographics**

**Query:**

SELECT

    d.AGE\_DESC,

    d.INCOME\_DESC,

    COUNT(DISTINCT t.BASKET\_ID) AS Total\_Orders,

    ROUND(SUM(t.SALES\_VALUE), 2) AS Total\_Revenue

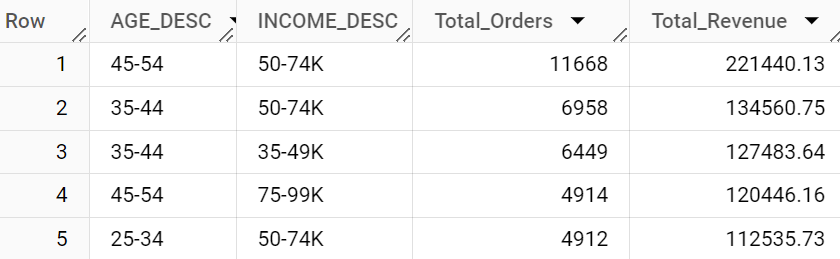
FROM ecommerce\_sql.transaction\_data t

JOIN ecommerce\_sql.demographic d ON t.household\_key = d.household\_key

GROUP BY 1,2

ORDER BY 4 DESC;

**Result:**

****

**Insights:**

* Customers between age group 45-54 with income 50-74K are the ones generating highest revenue

**Recommendations:** Customer outreach program might help with retaining them like loyalty points etc.

**VI. Find the number of orders that have small, medium or large order value (small:0-10 dollars, medium:10-20 dollars, large:20+)**

**Query:**

WITH sales\_category AS

(

    SELECT basket\_id, ROUND(SUM(sales\_value), 2) AS Sales

    FROM ecommerce\_sql.transaction\_data

    GROUP BY 1

)

SELECT

    CASE

        WHEN Sales BETWEEN 0 AND 10 THEN 'Small'

        WHEN Sales > 10 AND Sales <= 20 THEN 'Medium'

        WHEN Sales > 20 THEN 'Large'

    END AS Order\_Category,

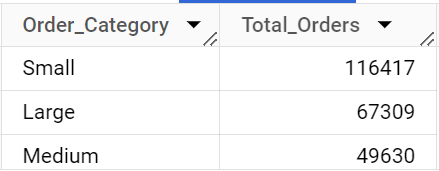
    COUNT(basket\_id) AS Total\_Orders

FROM sales\_category

GROUP BY 1

ORDER BY 2 DESC;

**Result:**

****

**Insights:**

* Almost 50% orders are from Small Order Category.
* Have good proportion of orders with Large Category as well

**Recommendations:** Attract customers with less order value to make them increase in order size so that the Average order value increases in turn increases the profits

**VII. Find top 3 stores with highest foot traffic for each week (Foot traffic: number of customers transacting)**

**Query:**

WITH store\_traffic AS (

    SELECT WEEK\_NO, STORE\_ID, COUNT(DISTINCT household\_key) AS foot\_traffic

    FROM ecommerce\_sql.transaction\_data

    GROUP BY 1,2

),

rank\_traffic AS (

    SELECT WEEK\_NO, STORE\_ID, foot\_traffic,

           RANK() OVER (PARTITION BY WEEK\_NO ORDER BY foot\_traffic DESC) AS pos

    FROM store\_traffic

)

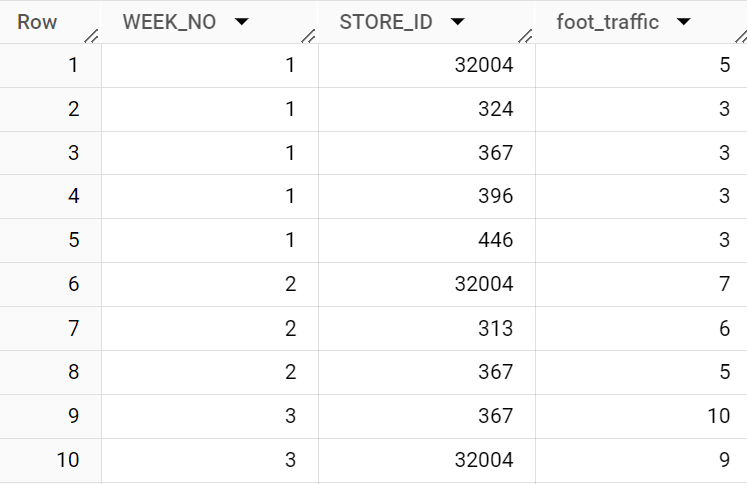
SELECT WEEK\_NO, STORE\_ID, foot\_traffic

FROM rank\_traffic

WHERE pos <= 3

ORDER BY 1,3 DESC,2;

**Result:**

****

**VIII. Create a basic customer profiling with first, last visit, number of visits, average money spent per visit and total money spent order by highest avg money**

**Query:**

SELECT

    household\_key,

    MIN(DAY) AS first\_visit,

    MAX(DAY) AS last\_visit,

    COUNT(DISTINCT BASKET\_ID) AS num\_visits,

    round(SUM(SALES\_VALUE)/COUNT(DISTINCT BASKET\_ID),2) AS avg\_money\_spent\_per\_visit,

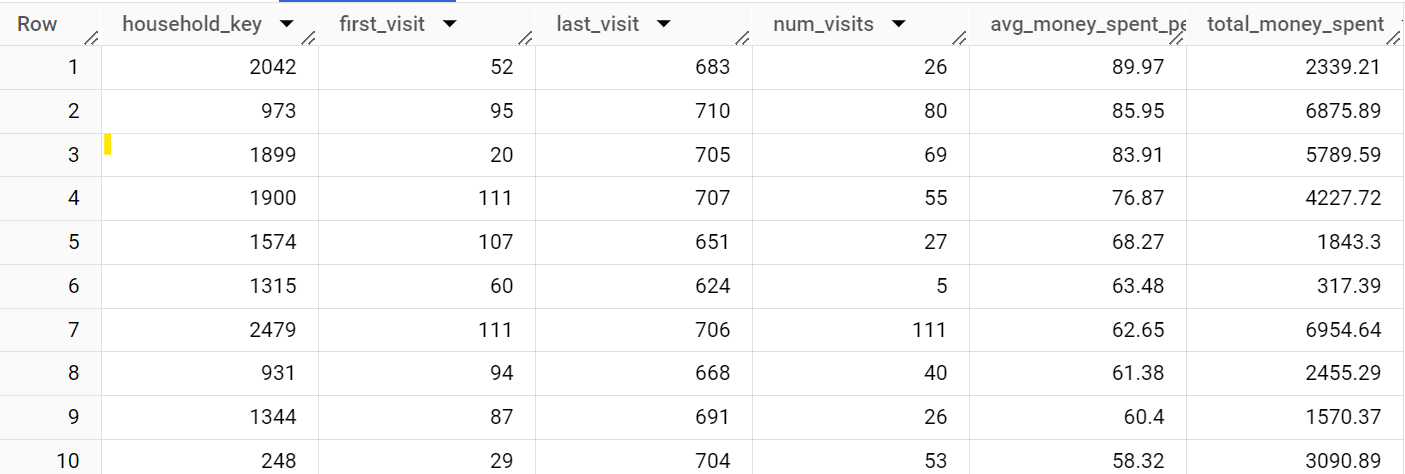
    round(SUM(SALES\_VALUE),2) AS total\_money\_spent

FROM ecommerce\_sql.transaction\_data

GROUP BY household\_key

ORDER BY avg\_money\_spent\_per\_visit DESC;

**Result:**

****

**IX. Do a single customer analysis selecting most spending customer for whom we have demographic information(because not all customers in transaction data are present in demographic table)(show the demographic as well as total spent)**

**Query:**

WITH high\_spending AS (

    SELECT household\_key, ROUND(SUM(sales\_value),2) AS total\_spent

    FROM ecommerce\_sql.transaction\_data

    GROUP BY 1

    ORDER BY 2 DESC

)

SELECT d.\*, hs.total\_spent

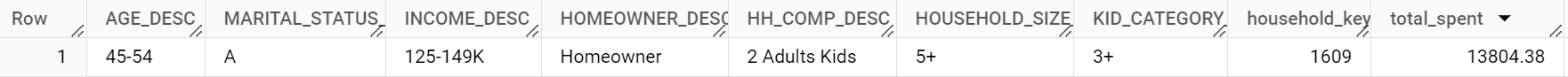
FROM ecommerce\_sql.demographic d

JOIN high\_spending hs ON d.household\_key = hs.household\_key

ORDER BY hs.total\_spent desc

LIMIT 1;

**Result:**

****

**X. Find products (product table: SUB\_COMMODITY\_DESC) which are most frequently bought together and the count of each combination bought together. do not print a combination twice (A-B / B-A)**

**Query:**

WITH cte1 AS (

    SELECT td.BASKET\_ID, p.SUB\_COMMODITY\_DESC AS product\_desc

    FROM ecommerce\_sql.transaction\_data td

    JOIN ecommerce\_sql.product p

      ON td.PRODUCT\_ID = p.PRODUCT\_ID

),

product\_combinations AS (

    SELECT bp1.BASKET\_ID, bp1.product\_desc AS product\_A, bp2.product\_desc AS product\_B

    FROM cte1 bp1

    JOIN cte1 bp2

        ON bp1.BASKET\_ID = bp2.BASKET\_ID

        AND bp1.product\_desc < bp2.product\_desc

)

SELECT product\_A, product\_B, COUNT(\*) AS frequency

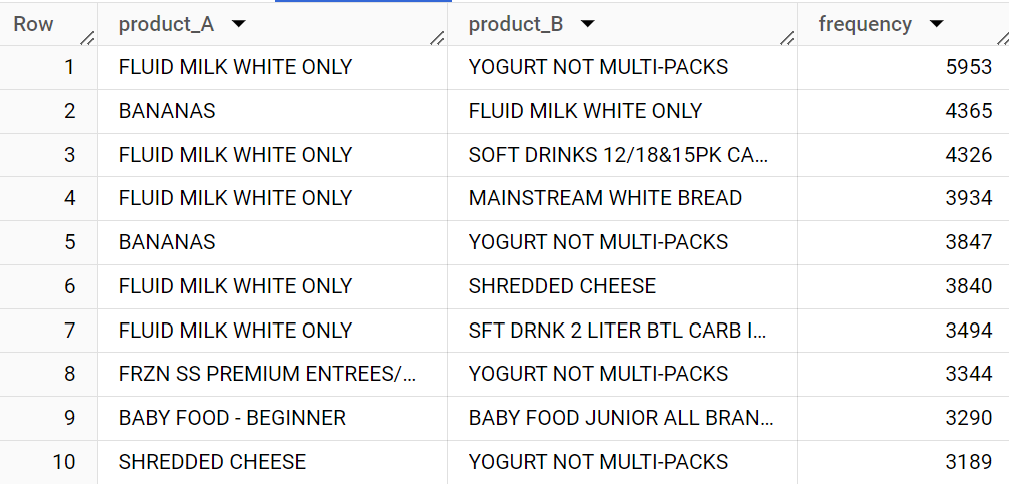
FROM product\_combinations

GROUP BY product\_A, product\_B

ORDER BY frequency DESC

LIMIT 10;

**Result:**

****

**Insights:**

* Fluid White Milk is the product that was brought together with other products well

**Recommendations:**

* Optimise the search Engine/ place the products that are frequently brought together to increase the sales
* Product placement of top selling products to be done so that they can catch the customer’s eye

**XI. Find the weekly change in Revenue Per Account (RPA) (difference in spending by each customer compared to last week)(use lag function)**

**Query:**

WITH weekly\_rev AS (

    SELECT household\_key, WEEK\_NO, ROUND(SUM(SALES\_VALUE),2) AS weekly\_spend

    FROM ecommerce\_sql.transaction\_data

    GROUP BY 1,2

),

weekly\_rpa AS (

    SELECT household\_key, week\_no, weekly\_spend,

           LAG(weekly\_spend) OVER (PARTITION BY household\_key ORDER BY WEEK\_NO) AS previous\_week\_spend

    FROM weekly\_rev

)

SELECT household\_key, week\_no, previous\_week\_spend,weekly\_spend,

       ROUND(weekly\_spend - previous\_week\_spend,2) AS rpa\_change

FROM weekly\_rpa

ORDER BY 1;

**Result:**

****