SALES DATA ANALYSIS USING SQL

Initial Data Cleaning: -

Handling null values=

```
UPDATE Salesdata.Products
SET
  PRODUCT_ID=COALESCE(PRODUCT_ID, null),
  MANUFACTURER=COALESCE(MANUFACTURER, null),
  DEPARTMENT=COALESCE(DEPARTMENT, 'Not Available'),
  BRAND=COALESCE(BRAND, 'Not Available'),
  COMMODITY_DESC=COALESCE(COMMODITY_DESC, 'Not Available'),
  SUB_COMMODITY_DESC=COALESCE(SUB_COMMODITY_DESC, 'Not Available'),
  CURR_SIZE_OF_PRODUCT=COALESCE(CURR_SIZE_OF_PRODUCT, 'Not Available');
```

```
UPDATE Salesdata.Transactions
SET
household_key=COALESCE(household_key, null),
BASKET_ID=COALESCE(BASKET_ID, null),
DAY=COALESCE(DAY, null),
PRODUCT_ID=COALESCE(PRODUCT_ID, null),
QUANTITY=COALESCE(QUANTITY, null),
SALES_VALUE=COALESCE(SALES_VALUE, null),
STORE_ID=COALESCE(STORE_ID, null),
RETAIL_DISC=COALESCE(RETAIL_DISC, null),
TRANS_TIME=COALESCE(TRANS_TIME, null),
WEEK_NO=COALESCE(WEEK_NO, null),
COUPON_DISC=COALESCE(COUPON_DISC, null),
COUPON_MATCH_DISC=COALESCE(COUPON_MATCH_DISC, null);
```

```
UPDATE Salesdata.Demographics
SET
AGE_DESC=COALESCE(AGE_DESC, 'Not Available'),
MARITAL_STATUS_CODE=COALESCE(MARITAL_STATUS_CODE, 'Not Available'),
INCOME_DESC=COALESCE(INCOME_DESC, 'Not Available'),
HOMEOWNER_DESC=COALESCE(HOMEOWNER_DESC, 'Not Available'),
HH_COMP_DESC=COALESCE(HH_COMP_DESC, 'Not Available'),
HOUSEHOLD_SIZE_DESC=COALESCE(HOUSEHOLD_SIZE_DESC, 'Not Available'),
KID_CATEGORY_DESC=COALESCE(KID_CATEGORY_DESC, 'Not Available'),
household_key=COALESCE(household_key, null);
```

Note: - These above need to be done in order to increase the data quality but here since I used sandbox version of bigquery, so couldn't perform but this is good practice to be done.

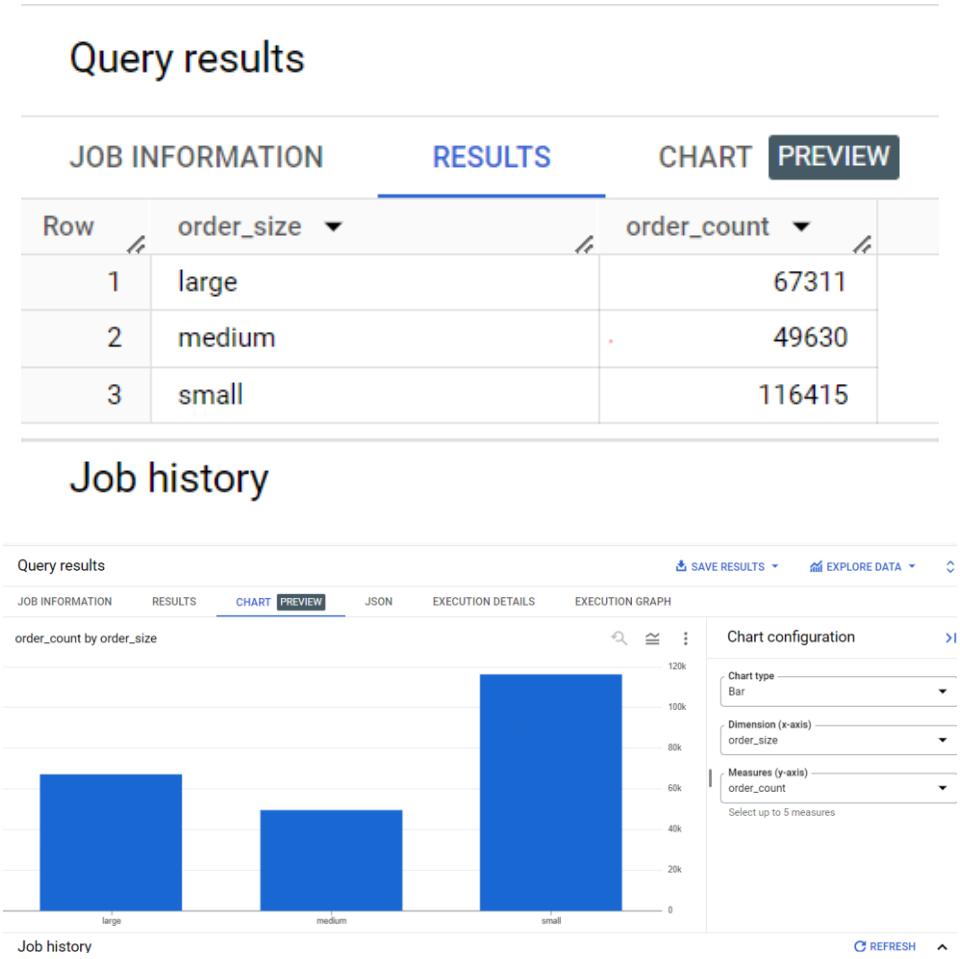
Mandatory Questions

Problem Statement 1: *Find the number of orders that have small, medium or large order value (small:0-10 dollars, medium:10-20 dollars, large:20+)*

Script: -

```sql
WITH Ordervalues AS (
  SELECT
    t.BASKET_ID,
    SUM(t.SALES_VALUE) AS total_order_value
  FROM
    Salesdata.Transactions t
  GROUP BY
    t.BASKET_ID
)
SELECT
  CASE
    WHEN total_order_value >= 0 AND total_order_value <= 10 THEN 'small'
    WHEN total_order_value > 10 AND total_order_value <= 20 THEN 'medium'
    WHEN total_order_value > 20 THEN 'large'
    ELSE 'unknown'
  END AS order_size,
  COUNT(DISTINCT ov.BASKET_ID) AS order_count
FROM
  Ordervalues ov
GROUP BY
  order_size
ORDER BY
  order_size;
```

Output: -



Insights: -

The observation reveals a pattern in the order distribution (based on order values) :

It shows that small orders, defined with order values between 0 and 10 dollars, are most frequently preferred followed by large orders (values exceeding 20 dollars), and finally medium orders (values between 10 and 20 dollars).

Problem Statement 2: Find the number of orders that are small, medium or large order value(small:0-5 dollars, medium:5-10 dollars, large:10+)

Script: -

```
WITH OrderValueRanges AS (
  SELECT
    t.BASKET_ID,
    SUM(t.SALES_VALUE) AS order_value
  FROM
    Salesdata.Transactions t
  GROUP BY
    t.BASKET_ID
)
SELECT
  CASE
    WHEN order_value >= 0 AND order_value < 5 THEN 'Small'
    WHEN order_value >= 5 AND order_value <= 10 THEN 'Medium'
    WHEN order_value > 10 THEN 'Large'
  END AS order_size,
  COUNT(DISTINCT ov.BASKET_ID) AS order_count
FROM
  OrderValueRanges ov
GROUP BY
  order_size
ORDER BY
  order_size;
```

Output: -

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON

Row	order_size	order_count
1	Large	116941
2	Medium	46663
3	Small	69752

Insights: -

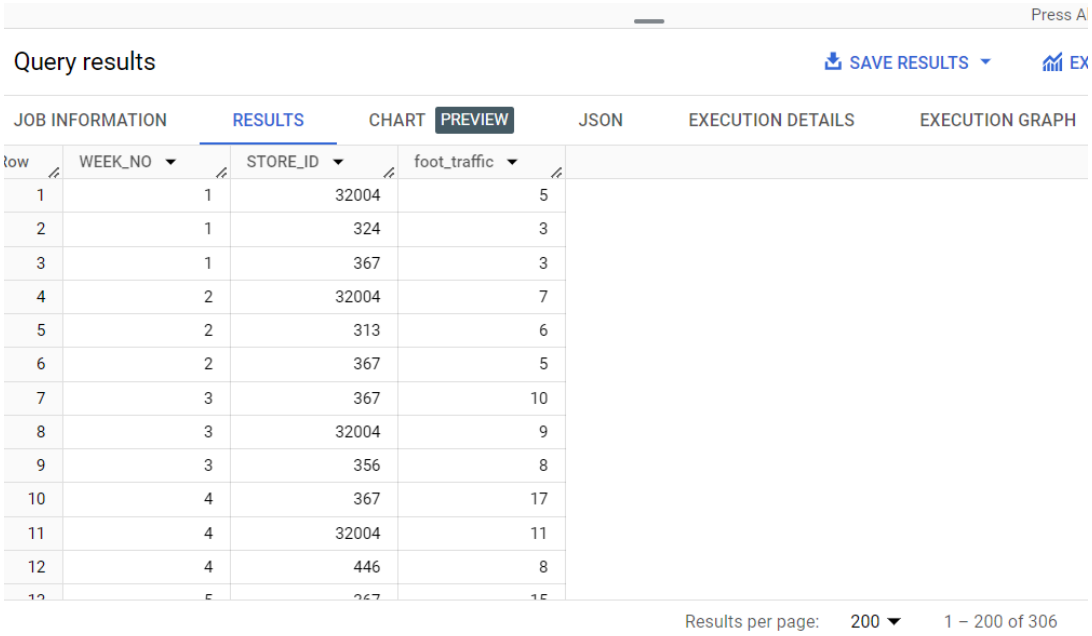
It's seen that maximum orders are done with large order value ($10+) followed by small and finally medium. Therefore, we can conclude that large product items are preferred more by customer here as compared to problem statement 1 where small was preferred more (bracket values of order size difference, led to 2 different observations as compared to question 1 and 2).

Problem Statement 3: Find top 3 stores with highest foot traffic for each week (Foot traffic: number of customers transacting)

Script: -

```
SELECT
  WEEK_NO,
  STORE_ID,
  foot_traffic
FROM (
  SELECT
    WEEK_NO,
    STORE_ID,
    foot_traffic,
    ROW_NUMBER() OVER (PARTITION BY WEEK_NO ORDER BY foot_traffic DESC) AS
store_rank
  FROM (
    SELECT
      t.WEEK_NO,
      t.STORE_ID,
      COUNT(DISTINCT t.household_key) AS foot_traffic
    FROM
       Salesdata.Transactions t
    GROUP BY
      t.WEEK_NO, t.STORE_ID
  ) AS StoreFootTraffic
) AS RankedStores
WHERE
  store_rank <= 3
ORDER BY
  WEEK_NO, store_rank;
```

Output: -



Row	WEEK_NO	STORE_ID	foot_traffic
1	1	32004	5
2	1	324	3
3	1	367	3
4	2	32004	7
5	2	313	6
6	2	367	5
7	3	367	10
8	3	32004	9
9	3	356	8
10	4	367	17
11	4	32004	11
12	4	446	8

Insights: -

The above analysis of weekly foot traffic in the top 3 stores unveils crucial patterns in customer behavior,

Store id : 32004 attract the highest number of customers each week

The 2nd graph shows on week 17th with max foot traffic of 54.

By this, We can adapt marketing efforts, inventory management, and customer service resources accordingly.

Further, This will aid in tailoring business strategies to enhance customer engagement and maximize revenue.

Problem Statement 4: 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

Script: -

```sql
WITH CustomerVisits AS (
  SELECT
    t.household_key,
    MIN(t.DAY) AS first_visit,
    MAX(t.DAY) AS last_visit,
    COUNT(DISTINCT t.BASKET_ID) AS num_visits,
    AVG(t.SALES_VALUE) AS avg_money_spent_per_visit,
    SUM(t.SALES_VALUE) AS total_money_spent
  FROM
    Salesdata.Transactions t
  GROUP BY
    t.household_key
)
SELECT
  c.household_key,
  c.first_visit,
  c.last_visit,
  c.num_visits,
  c.avg_money_spent_per_visit,
  c.total_money_spent,
  d.AGE_DESC,
  d.MARITAL_STATUS_CODE,
  d.INCOME_DESC,
  d.HOMEOWNER_DESC,
  d.HH_COMP_DESC,
  d.HOUSEHOLD_SIZE_DESC,
  d.KID_CATEGORY_DESC
FROM
  CustomerVisits c
JOIN
  Salesdata.Demographics d ON c.household_key = d.household_key
ORDER BY
  c.avg_money_spent_per_visit DESC;
```

Output: -

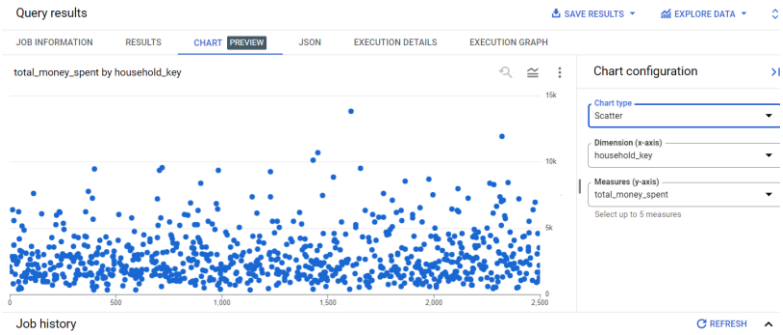
Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	first_visit	last_visit	num_visits	avg_money_spent_pe	total_money_spent	AGE_DESC	MARITAL_STATUS_CODE
1	755	36	709	201	9.48	5461.54	35-44	U
2	1357	59	710	77	6.62	1688.05	55-64	A
3	2162	98	710	139	6.23	7259.18	45-54	U
4	2097	26	645	175	6.23	4425.73	35-44	A
5	101	116	708	168	6.23	3632.15	45-54	A
6	853	65	707	454	5.59	6892.11	25-34	B
7	513	21	705	156	5.57	6040.48	45-54	U
8	2203	42	711	93	5.49	2748.54	45-54	A
9	13	101	709	201	5.47	6416.3	25-34	U
10	193	32	694	100	5.35	2398.54	45-54	A
11	671	77	694	101	5.35	2536.56	35-44	U
12	1935	114	711	133	5.22	6531.94	55-64	A

Results per page: 50 1 – 50 of 801



Insights: -

Understanding the age, marital status, and income bracket of this customer provides a comprehensive overview of customer behaviors and spending patterns; the above analysis showcases:

- Notably, Customer 1, aged 35-44, exhibits a high level of loyalty, making a substantial 201 visits over the observed period. Despite an unknown marital status, this customer showcases a consistent average spending of $9.48 per visit, contributing to a remarkable total spending of $5461.54. This suggests an opportunity for targeted loyalty programs or personalized promotions to further enhance the relationship with this customer.
- Customer 2, aged 55-64 and married, represents another valuable segment. Despite a lower visit count of 77, this customer demonstrates a steady spending pattern with an average of $6.61 per visit, resulting in a total spending of $1688.05.

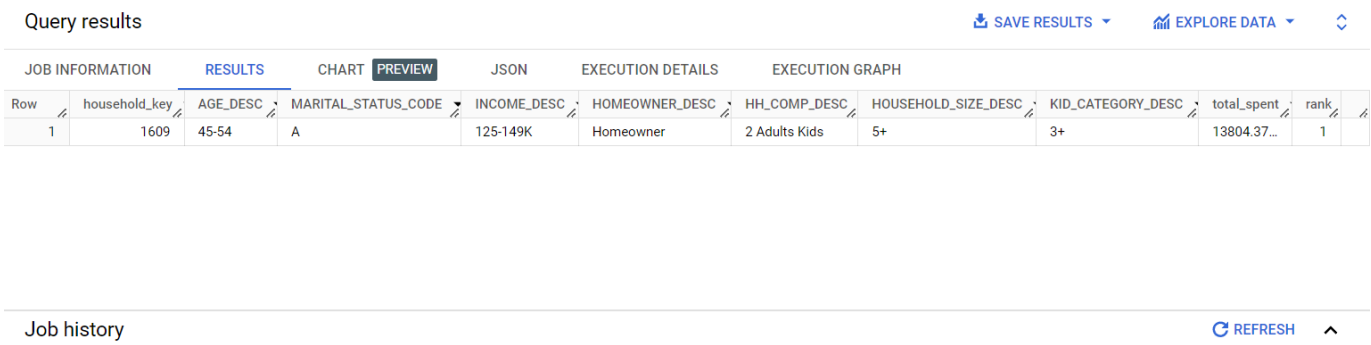
From last graph, it clearly visible that max customers fall under $0-5k in terms of purchase behavior. Also, the max purchase is done by customer id 1604 for an amount of $13.8k over period of time.

Problem Statement 5: 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)

Script: -

```
WITH CustomerSpending AS (
  SELECT
    t.household_key,
    d.AGE_DESC,
    d.MARITAL_STATUS_CODE,
    d.INCOME_DESC,
    d.HOMEOWNER_DESC,
    d.HH_COMP_DESC,
    d.HOUSEHOLD_SIZE_DESC,
    d.KID_CATEGORY_DESC,
    SUM(t.SALES_VALUE) AS total_spent
  FROM
    Salesdata.Transactions t
  JOIN
    Salesdata.Demographics d ON t.household_key = d.household_key
  GROUP BY
    t.household_key, d.AGE_DESC, d.MARITAL_STATUS_CODE, d.INCOME_DESC,
    d.HOMEOWNER_DESC, d.HH_COMP_DESC, d.HOUSEHOLD_SIZE_DESC, d.KID_CATEGORY_DESC
)
SELECT
  cs.*,
  ROW_NUMBER() OVER (ORDER BY cs.total_spent DESC) AS rank
FROM
  CustomerSpending cs
WHERE
  cs.household_key IS NOT NULL
LIMIT 1;
```

Output: -



Insights: -

The single customer analysis, focusing on the most spending customer with available demographic information, is as follows:

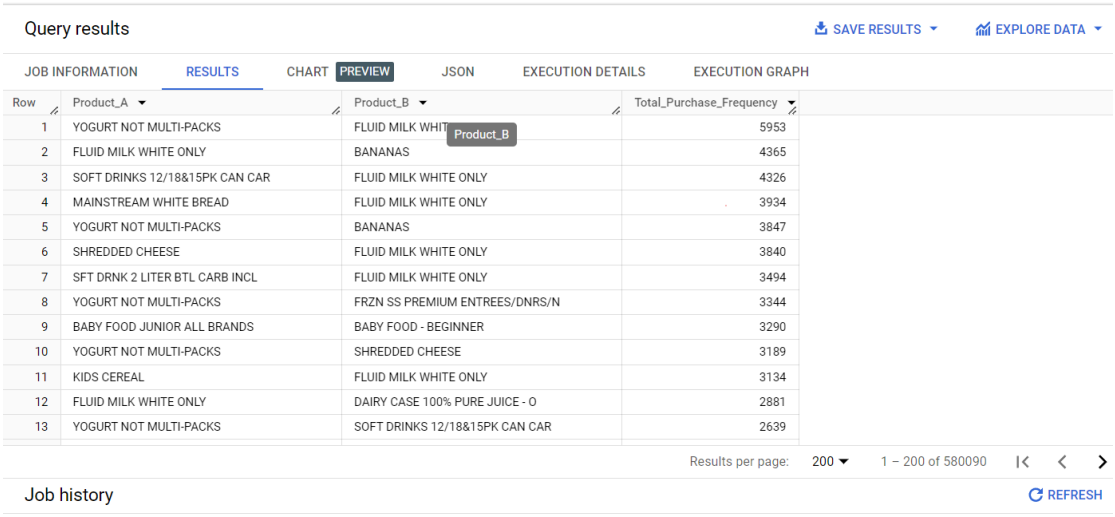
- household key 1609 emerges as the most significant spender within the dataset.
- This customer, falling within the 45-54 age bracket, identified as married with a substantial income bracket ranging from $125,000 to $149,000, being a homeowner and having a household composition of 2 Adult kids further characterizes this customer's demographic profile.

Problem Statement 6: 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)

Script: -

```sql
WITH ProductTemp AS (
  SELECT
    CASE WHEN p1.SUB_COMMODITY_DESC > p2.SUB_COMMODITY_DESC THEN p1.SUB_COMMOD-
ITY_DESC ELSE p2.SUB_COMMODITY_DESC END AS Product_A,
    CASE WHEN p1.SUB_COMMODITY_DESC > p2.SUB_COMMODITY_DESC THEN p2.SUB_COMMOD-
ITY_DESC ELSE p1.SUB_COMMODITY_DESC END AS Product_B,
    COUNT(*) AS Frequency_of_Purchase
  FROM
    Salesdata.Transactions t1
  JOIN
    Salesdata.Transactions t2
    ON t1.household_key = t2.household_key
    AND t1.BASKET_ID = t2.BASKET_ID
    AND t1.PRODUCT_ID > t2.PRODUCT_ID
  JOIN
    Salesdata.Products p1
    ON t1.PRODUCT_ID = p1.PRODUCT_ID
  JOIN
    Salesdata.Products p2
    ON t2.PRODUCT_ID = p2.PRODUCT_ID
  GROUP BY
    Product_A, Product_B
)
SELECT
  Product_A,
  Product_B,
  SUM(Frequency_of_Purchase) AS Total_Purchase_Frequency
FROM
  ProductTemp
WHERE
  Product_A != Product_B
GROUP BY
  Product_A, Product_B
ORDER BY
  Total_Purchase_Frequency DESC;
```

Output: -



Insights: -

The combination of “FLUID MILK WHITE ONLY” and “YOGURT NOT MULTI-PACKS” is purchased 5953 times and is the highest purchased combo as compared to rest of combos. Same analysis pattern goes for the rest of the products (arranged in desc format of frequency of purchase count).

Problem Statement 7: Find the weekly change in Revenue Per Account (RPA) (difference in spending by each customer compared to last week) (use lag function)

Script: -

```
WITH WeeklyRevenue AS (
  SELECT
    t.household_key,
    t.WEEK_NO,
    round(SUM(t.SALES_VALUE),2) AS weekly_revenue
  FROM
    Salesdata.Transactions t
  GROUP BY
    t.household_key, t.WEEK_NO
)
SELECT
  t.household_key,
  t.WEEK_NO,
  t.weekly_revenue,
  round(LAG(t.weekly_revenue) OVER (PARTITION BY t.household_key ORDER BY
t.WEEK_NO),2) AS prev_weekly_revenue,
  round(COALESCE(t.weekly_revenue - LAG(t.weekly_revenue) OVER (PARTITION BY
t.household_key ORDER BY t.WEEK_NO), t.weekly_revenue),2) AS weekly_change_in_rpa
FROM
  WeeklyRevenue t
ORDER BY
  t.household_key, t.WEEK_NO;
```

Output: -

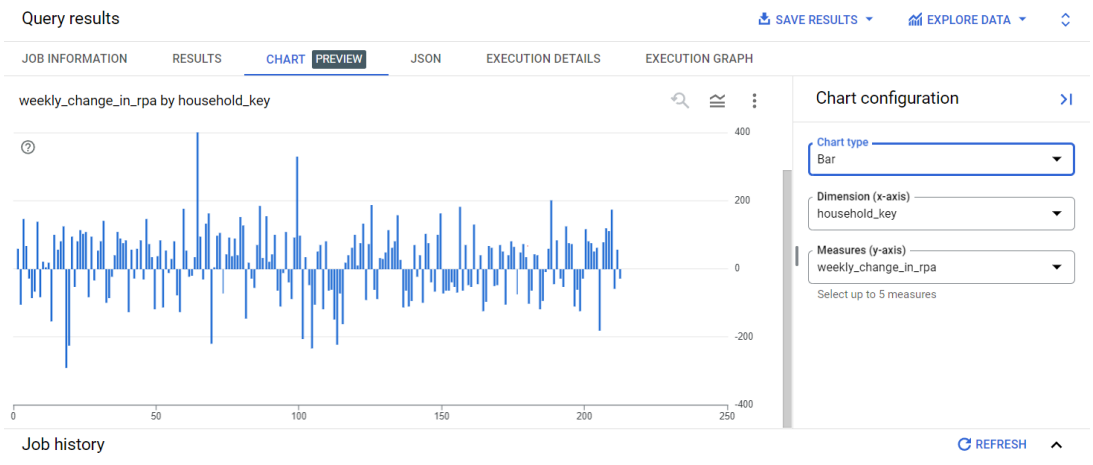
Query results

SAVE RESULTS | EXP

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	WEEK_NO	weekly_revenue	prev_weekly_revenue	weekly_change_in_rpa
1	1	8	42.58	null	42.58
2	1	10	14.01	42.58	-28.57
3	1	13	14.03	14.01	0.02
4	1	14	25.71	14.03	11.68
5	1	15	10.98	25.71	-14.73
6	1	16	9.09	10.98	-1.89
7	1	17	13.98	9.09	4.89
8	1	19	47.35	13.98	33.37
9	1	20	31.77	47.35	-15.58
10	1	22	38.98	31.77	7.21
11	1	23	26.36	38.98	-12.62
12	1	24	35.13	26.36	8.77
13	1	25	17.23	35.13	-17.9

Results per page: 200 1 – 200 of 117599



Insights: -

The analysis of the weekly change in Revenue Per Account (RPA) utilizing the lag function

- unveils valuable insights into the spending dynamics of each customer over time.
- By comparing the spending for each customer with the previous week, it becomes evident that some customers exhibit a negative difference in spending for certain week numbers suggesting fluctuations in customer spending behaviors.

Self-Prepared Questions

Problem Statement 1: Analyze the impact of sales by considering factors such as time of day, customer demographics, etc.

Script: -

```
SELECT
    t.TRANS_TIME,
    t.DAY,
    d.AGE_DESC,
    d.INCOME_DESC,
    p.SUB_COMMODITY_DESC,
    AVG(SALES_VALUE) AS avg_sales_value,
    CASE
        WHEN t.TRANS_TIME BETWEEN 0000 AND 559 THEN 'Night'
        WHEN t.TRANS_TIME BETWEEN 600 AND 1159 THEN 'Morning'
        WHEN t.TRANS_TIME BETWEEN 1200 AND 1759 THEN 'Afternoon'
        WHEN t.TRANS_TIME BETWEEN 1800 AND 2359 THEN 'Evening'
        ELSE 'Unknown'
    END AS time_of_day
FROM Salesdata.Transactions t
JOIN Salesdata.Products p ON t.PRODUCT_ID = p.PRODUCT_ID
JOIN Salesdata.Demographics d ON t.household_key = d.household_key
GROUP BY
    t.TRANS_TIME,
    t.DAY,
    d.AGE_DESC,
    d.INCOME_DESC,
    p.SUB_COMMODITY_DESC,
    time_of_day
ORDER BY avg_sales_value DESC;
```

Output: -

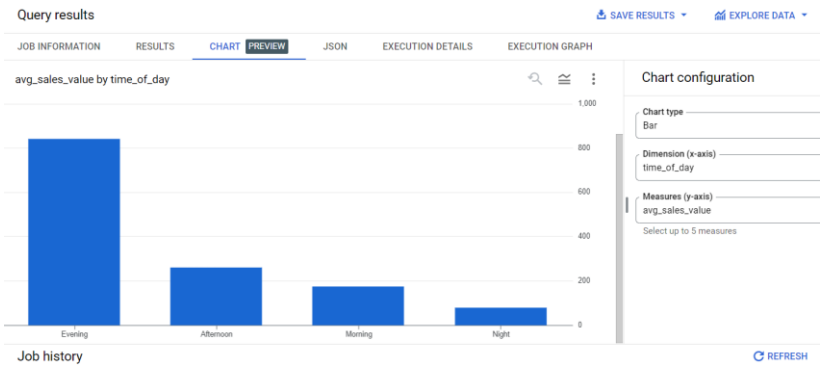
Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	TRANS_TIME	DAY	AGE_DESC	INCOME_DESC	SUB_COMMODITY_DESC	avg_sales_value	time_of_day
1	2038	339	45-54	125-149K	ELECTRONIC GIFT CARDS ACT...	840.0	Evening
2	1643	530	55-64	15-24K	NO SUBCOMMODITY DESCRIP...	261.58	Afternoon
3	1314	642	55-64	100-124K	ELECTRONIC GIFT CARDS REF...	250.0	Afternoon
4	1554	284	35-44	Under 15K	NO SUBCOMMODITY DESCRIP...	246.6	Afternoon
5	1606	181	45-54	100-124K	GASOLINE-REG UNLEADED	210.0	Afternoon
6	1734	127	65+	75-99K	OUTSIDE VENDORS GIFT CARDS	209.94	Afternoon
7	1311	272	65+	75-99K	OUTSIDE VENDORS GIFT CARDS	179.91	Afternoon
8	941	97	45-54	50-74K	SOFT DRINKS 12/18&15PK CA...	176.03	Morning
9	1006	432	25-34	35-49K	GAS GRILLS	169.99	Morning
10	1341	466	45-54	125-149K	PORK-FULLY COOKED	167.88	Afternoon
11	1509	219	35-44	35-49K	TICKETS	155.6	Afternoon
12	2045	273	19-24	35-49K	ELECTRONIC GIFT CARDS ACT...	150.0	Evening
13	1317	645	35-44	35-49K	OUTSIDE VENDORS GIFT CARDS	150.0	Afternoon

Results per page: 200 | 1 – 200 of 648822



Insights: -

The highest average sales value is $840 for age group 45-44 on 339th day at time 20:38 for electronic gift cards. The same logic goes for rest of the records.

From 2nd pic, its seen that average sales are higher during evening time followed by afternoon, etc.

Problem Statement 2: Calculate the average transaction value over time to understand if there are any significant changes in customer spending habits.

Script: -

```
SELECT
    WEEK_NO,
    round(AVG(SALES_VALUE),2) AS avg_transaction_value
FROM Salesdata.Transactions
GROUP BY WEEK_NO
ORDER BY WEEK_NO;
```

Output: -

Query results

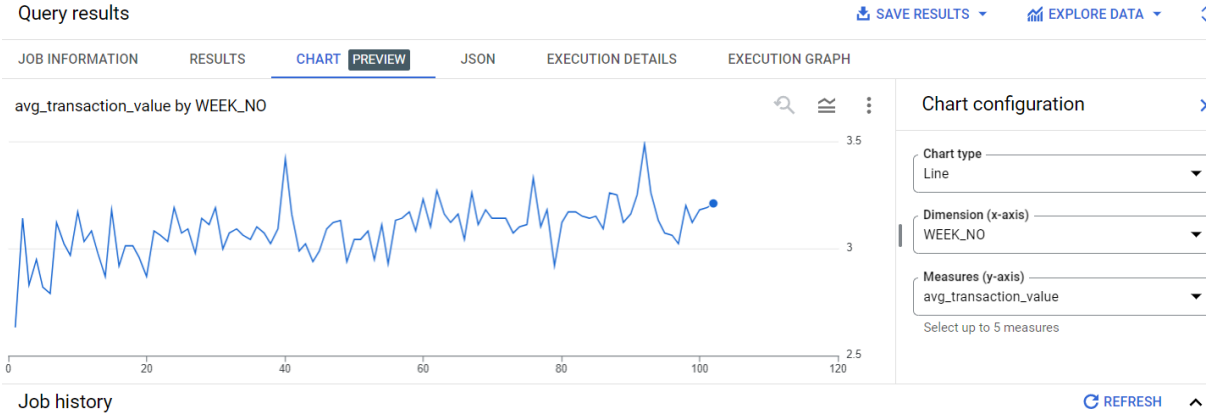
SAVE RESULTS EXPLORE DAT

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	WEEK_NO	avg_transaction_value
1	1	2.63
2	2	3.14
3	3	2.83
4	4	2.95
5	5	2.82
6	6	2.79
7	7	3.12
8	8	3.02
9	9	2.97
10	10	3.17
11	11	3.03

Results per page: 200 1 – 102 of 102

Job historv REFI



Insights: -

The average transaction value for week number 1 is $ 2.63 and same goes for rest of the week numbers. These can be further used to track down how the average transaction values changes across week number which further can lead to optimizing business strategies.

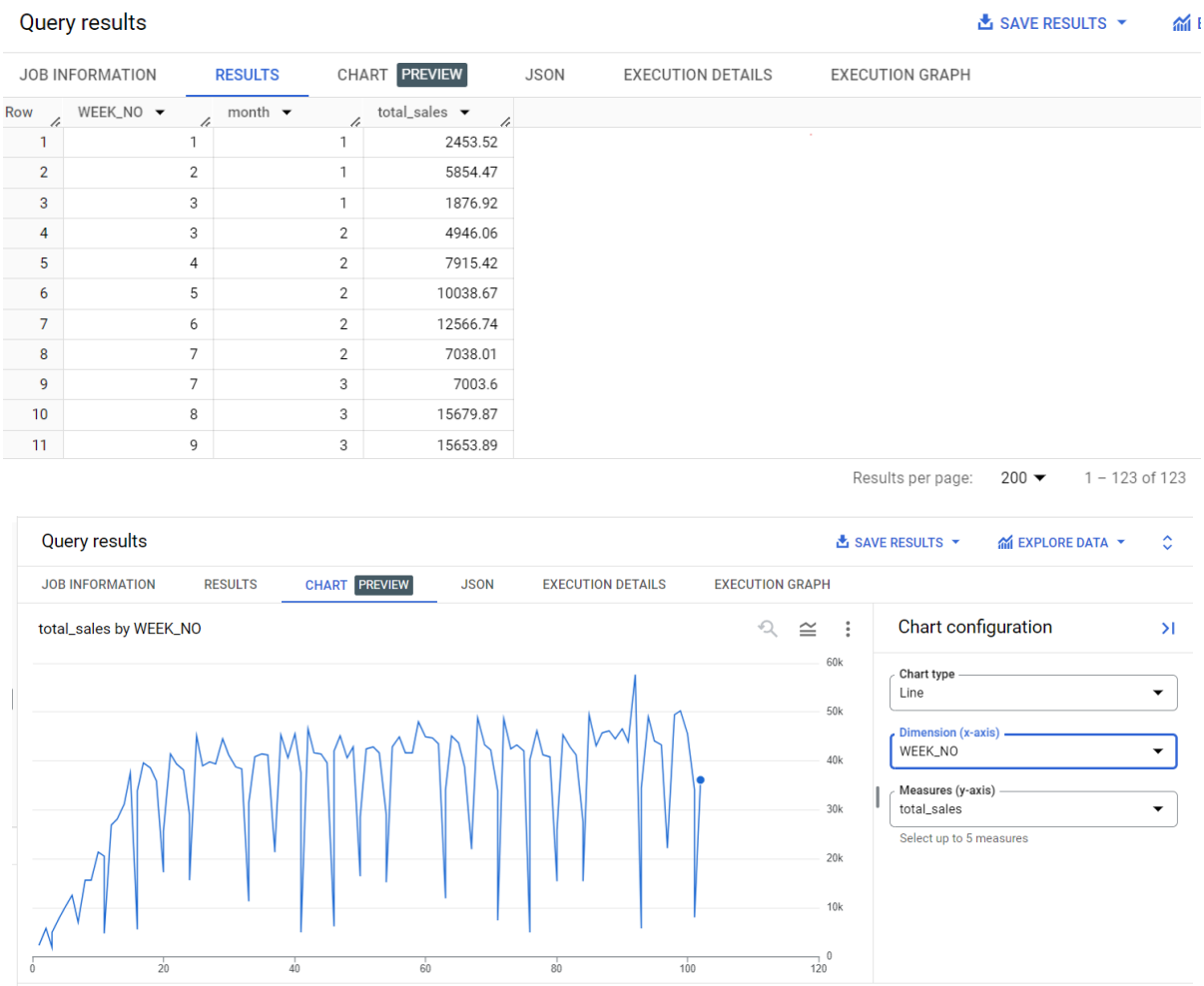
Problem Statement 3: Examine sales trends across different months/week to identify which months/week are more popular during specific times of the year

Case 1: - Month and Week number-based analysis

Script: -

```
SELECT
    WEEK_NO,
    CAST(DAY / 30 AS INT64) + 1 AS month,
    round(SUM(SALES_VALUE),2) AS total_sales
FROM Salesdata.Transactions
GROUP BY WEEK_NO, month
ORDER BY WEEK_NO, month;
```

Output: -



Insights: -

The month number 1 has 3 week numbers and for week number 1 it has total sales of $2453.52 and same trend goes for the rest of the week numbers. Also the peak is observed on 92nd week number with a total sales value of $57.72k (2nd graph).

Problem Statement 4: Dynamically segmenting customers based on their purchasing behavior

Script: -

```
SELECT
    household_key,
    MAX(DAY) AS last_purchase_date,
    COUNT(DISTINCT BASKET_ID) AS num_visits,
    round(AVG(SALES_VALUE),2) AS avg_spend,
    COUNT(DISTINCT PRODUCT_ID) AS num_unique_products
FROM Salesdata.Transactions
GROUP BY household_key
ORDER BY household_key;
```

Output: -

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	last_purchase_date	num_visits	avg_spend	num_unique_products
1	1	706	78	2.43	422
2	2	668	44	2.97	299
3	3	703	46	3.02	312
4	4	627	29	4.17	98
5	5	703	34	3.73	102
6	6	707	224	3.28	499
7	7	709	56	2.7	491
8	8	706	110	2.89	736
9	9	689	18	3.71	99
10	10	685	9	3.24	33
11	11	412	5	2.23	9

Results per page: 200 1 – 200 of 2500

Insights: -

For the household key 1 customer last latest purchase date was 706, total of 78 transactions he has done till date and average spend is $2.43 with total of 422 unique products (unique sub commodities brought till recent date). The same trend goes for rest of the household keys.

Problem Statement 5: Examine how customers purchase patterns change over time. This could include identifying seasonality or understanding if there are certain days of the week when specific products are more popular

Script: -

```
SELECT
    household_key,
    DAY,
    COUNT(*) AS num_purchases
FROM Salesdata.Transactions
GROUP BY household_key, DAY
ORDER BY household_key, DAY;
```

Output: -

Query results

SAVE RESULTS | EXPLO

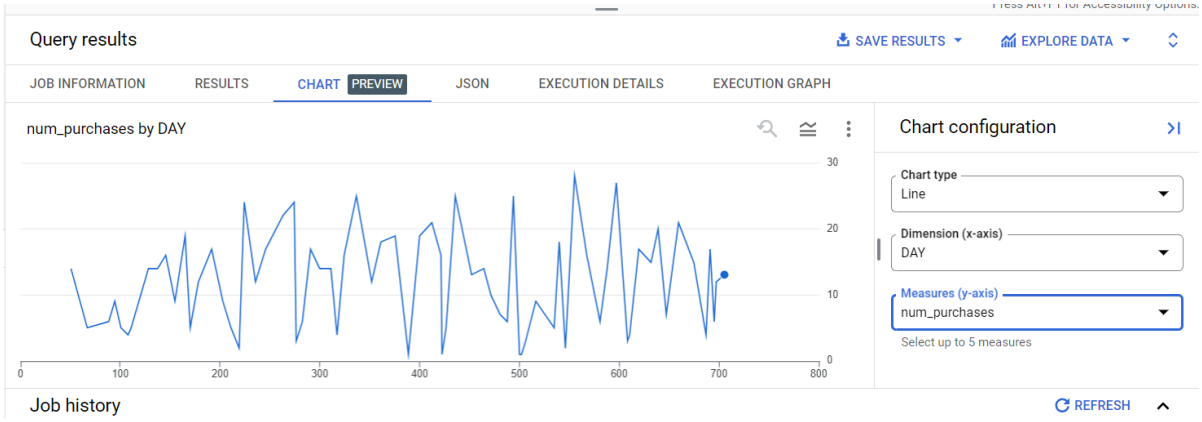
JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	DAY	num_purchases
1	1	51	14
2	1	67	5
3	1	88	6
4	1	94	9
5	1	101	5
6	1	108	4
7	1	111	5
8	1	128	14
9	1	137	14
10	1	146	16
11	1	155	9
12	1	165	19
13	1	170	5

Results per page: 50 | 1 – 50 of 201463

Insights: -

The above data shows how each customer (household key) have their number of purchases spanned across a particular day number. Here 1st row signifies that Day 1 has 14 purchases made by customer 1(household key). The same pattern goes for rest of customer and this data can help to build models/graphs to find a pattern which are the peak days of purchases for a customer and also how their purchase pattern changes over period of time.



- Also, if we see this graph (drilled down graph for household_key=1 only, we can infer the purchase pattern of the customer across different time span. The same insights can be generated for rest of customers using models and graphs to reduce churn rate)

Problem Statement 6: Tracking Brand Performance Across Demographics and Time and finding out the highest sales value, is incurred from which particular week number and how much is the total sales value.

Script: -

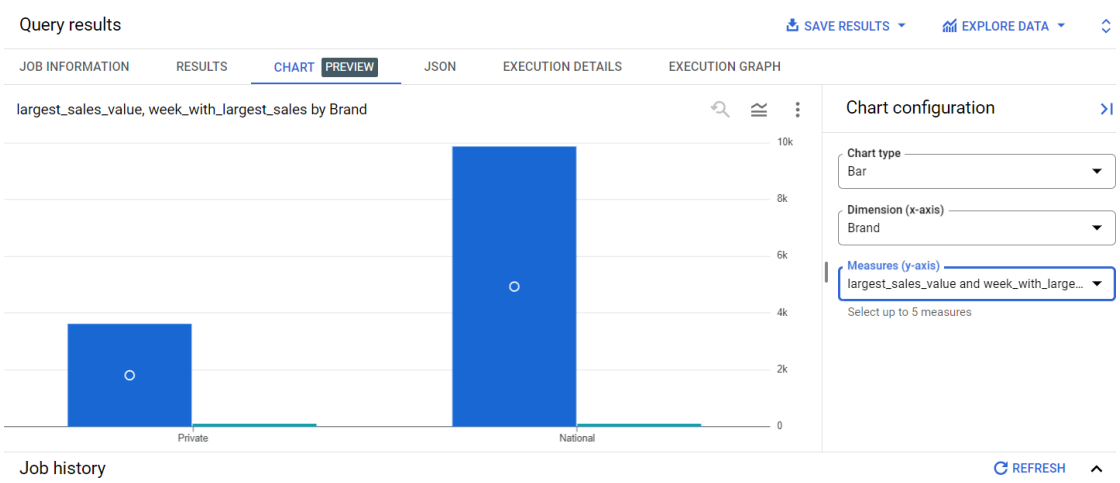
```sql
WITH WeeklyBrandSales AS (
    SELECT
        t.WEEK_NO,
        p.BRAND,
        d.AGE_DESC,
        SUM(t.SALES_VALUE) AS brand_sales
    FROM Salesdata.Transactions t
    JOIN Salesdata.Products p ON t.PRODUCT_ID = p.PRODUCT_ID
    JOIN Salesdata.Demographics d ON t.household_key = d.household_key
    GROUP BY t.WEEK_NO, p.BRAND, d.AGE_DESC
)
SELECT
    BRAND as Brand,
    round(MAX(brand_sales),2) AS largest_sales_value,
    MAX(WEEK_NO) AS week_with_largest_sales
FROM WeeklyBrandSales
GROUP BY BRAND;
```

Output: -

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS

Row	Brand	largest_sales_value	week_with_largest_sales
1	Private	3626.23	102
2	National	9856.48	102



Insights: -

The highest sales value w.r.t brand and week number are observed in 102nd week number for both brands. Out of the 2 brands, National brand has more weightage as compared to Private which signifies that overall customers prefer National brand.

Case 2: - Finding which brand plays a major role in the business.

Script: -

```sql
WITH TotalSales AS (
    SELECT SUM(SALES_VALUE) AS total_sales
    FROM Salesdata.Transactions
), BrandSales AS (
    SELECT
        p.BRAND,
        round(SUM(t.SALES_VALUE),2) AS brand_sales
    FROM Salesdata.Transactions t
    JOIN Salesdata.Products p ON t.PRODUCT_ID = p.PRODUCT_ID
    GROUP BY p.BRAND
)
SELECT
    b.BRAND as Brand,
    b.brand_sales as Total_Brand_Sales,
    round((b.brand_sales / a.total_sales) * 100,2) AS Brand_Sales_Percentage
FROM BrandSales b
JOIN TotalSales a ON 1=1;
```

Output: -

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | E

Row	Brand	Total_Brand_Sales	Brand_Sales_Percentage
1	Private	1119659.89	27.79
2	National	2909678.52	72.21

Insights: -

The highest sales value is incurred from National brand.

Problem Statement 7: Identify top-performing products and manufacturers to optimize inventory and promotions.

Script: -

```sql
SELECT
  p.MANUFACTURER,
  p.BRAND,
  p.COMMODITY_DESC,
  p.SUB_COMMODITY_DESC,
  round(SUM(t.SALES_VALUE),2) AS total_sales
FROM
  Salesdata.Transactions t
JOIN
  Salesdata.Products p ON t.PRODUCT_ID = p.PRODUCT_ID
GROUP BY
  p.MANUFACTURER,
  p.BRAND,
  p.COMMODITY_DESC,
  p.SUB_COMMODITY_DESC
ORDER BY
  total_sales DESC;
```

Output: -

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	MANUFACTURER	BRAND	COMMODITY_DESC	SUB_COMMODITY_DESC	total_sales
1	69	Private	COUPON/MISC ITEMS	GASOLINE-REG UNLEADED	301211.02
2	69	Private	FLUID MILK PRODUCTS	FLUID MILK WHITE ONLY	78619.04
3	1208	National	SOFT DRINKS	SOFT DRINKS 12/18&15PK CA...	34023.05
4	239	National	BEERS/ALES	BEERALEMALT LIQUORS	33137.29
5	539	National	CIGARETTES	CIGARETTES	31432.6
6	69	Private	CHEESE	SHREDDED CHEESE	25097.06
7	103	National	SOFT DRINKS	SOFT DRINKS 12/18&15PK CA...	24547.97
8	697	National	BEERS/ALES	BEERALEMALT LIQUORS	15840.08
9	2	National	TROPICAL FRUIT	BANANAS	15079.43
10	764	National	DIAPERS & DISPOSABLES	BABY DIAPERS	14982.66
11	69	Private	FUEL	GASOLINE-REG UNLEADED	14786.07
12	544	National	BAG SNACKS	TORTILLA/NACHO CHIPS	14780.6
13	4314	National	CHICKEN	CHICKEN BREAST BONELESS	14416.16

Results per page: 50 1 – 50 of 14093

Insights: -

The highest total sales value is observed for manufacturer 69 and it's a private brand and product sold is Gasoline. The same pattern goes for rest of the data extracted.

Problem Statement 8: Identify the top-performing stores based on sales

Script: -

```
SELECT
  t.STORE_ID,
  COUNT(DISTINCT t.BASKET_ID) AS total_baskets,
  round(SUM(t.SALES_VALUE),2) AS total_sales
FROM
  Salesdata.Transactions t
GROUP BY
  t.STORE_ID
ORDER BY
  total_sales DESC;
```

Output: -

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	STORE_ID	total_baskets	total_sales
1	367	6516	134105.46
2	406	4359	108814.66
3	361	4028	72493.72
4	429	2967	70752.61
5	343	4102	70265.6
6	356	3842	69026.02
7	375	3176	65788.5
8	381	4306	65400.58
9	292	3586	65201.6
10	31782	3734	61012.23
11	321	3497	59675.48
12	319	3597	59149.46
13	32004	3564	58666.02

Results per page: 50 1 – 50 of 552

Insights: -

The highest sales is made from store having store id 367 with total of 6516 orders which amounts to $134105.46.

Problem Statement 9: Segmenting customers based on their behavior and estimate their lifetime value

Script: -

```
WITH customer_behavior AS (
  SELECT
    t.household_key,
    COUNT(DISTINCT t.BASKET_ID) AS total_baskets,
    round(SUM(t.SALES_VALUE),2) AS total_sales
  FROM
    Salesdata.Transactions t
  GROUP BY
    t.household_key
)
SELECT
  cb.household_key,
  cb.total_baskets,
  cb.total_sales,
  d.AGE_DESC,
  d.INCOME_DESC,
  CASE
    WHEN cb.total_sales > 3000 THEN 'High Value'
    WHEN cb.total_sales > 1500 THEN 'Medium Value'
    ELSE 'Low Value'
  END AS customer_segment
FROM
  customer_behavior cb
JOIN
  Salesdata.Demographics d ON cb.household_key = d.household_key;
```

Output: -

Query results

SAVE RESULTS | EXPLOR

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	household_key	total_baskets	total_sales	AGE_DESC	INCOME_DESC	customer_segment
1	7	56	1733.88	45-54	50-74K	Medium Value
2	27	369	2814.89	45-54	25-34K	Medium Value
3	98	188	1250.37	35-44	35-49K	Low Value
4	117	93	1299.09	65+	35-49K	Low Value
5	131	301	3205.24	45-54	50-74K	High Value
6	219	212	5992.46	55-64	150-174K	High Value
7	248	53	3090.89	35-44	125-149K	High Value
8	294	230	2790.8	25-34	50-74K	Medium Value
9	304	254	4395.89	25-34	50-74K	High Value
10	367	100	3616.34	55-64	50-74K	High Value
11	451	132	1288.62	25-34	Under 15K	Low Value
12	454	333	4350.49	55-64	35-49K	High Value
13	542	105	1150.47	25-34	35-49K	Low Value

Results per page: 200 1 – 200 of 801

Insights: -

It aims to provide insights into customer behavior and estimate their lifetime value based on the number of total baskets, total sales, age, and income. Further this data can be used to target marketing, make retention strategies; provide product recommendations and make customer experience enhancement based on the “customer segment” column values.

Final Insights/Recommendations

1. Understanding order size sequencing (small, medium, large) implies potential upselling opportunities. Customers starting with small orders may lead to larger purchases, indicating possible loyalty. Tailor marketing strategies for each order size group to optimize revenue and enhance customer relationships.
2. Weekly analysis of top-performing stores provides valuable best practices for network-wide application. This insight aids strategic decision-making, optimizing operations, addressing location-specific challenges, and fostering a customer-centric approach for growth and satisfaction.
3. Profiling customers with the highest average spending per visit highlights segments with higher return potential. Focus marketing efforts, loyalty programs, and retention initiatives on these segments to optimize the customer experience and maximize revenue.
4. The financial significance of a high-spending customer emphasizes the value of catering to their specific needs. Tailor marketing strategies and promotions to enhance the overall customer experience, fostering lasting relationships with valuable segments.
5. Identifying frequently bought products together enhances recommendation algorithms and cross-selling strategies. Optimize product recommendations for a personalized shopping experience, unlocking revenue growth, and fostering customer loyalty.
6. Weekly RPA change highlights negative spending differences, aiding in investigating fluctuations. Proactive responses to shifts in customer behavior, influenced by seasonal trends or preferences, optimize revenue and enhance customer satisfaction.
7. While National brand dominates in sales, exploring additional aspects such as customer reviews, brand loyalty, and preferences to gain a holistic understanding beyond revenue figures can be useful.
8. Utilizing data on customer loyalty, demonstrated by consistent spending patterns over time, to develop targeted retention strategies. Implementing loyalty programs or exclusive promotions to enhance relationships with such customers, to promote growth.
9. Identifying unique product combinations beyond the top purchase combo. Promoting these combinations to encourage diverse purchasing behaviors and offering a wider array of options to customers.
10. Analyzing the frequency of peak purchase days identified through customer purchase patterns. Understanding if these peak days coincide with specific events, holidays, or promotional periods to optimize marketing strategies.
11. Leveraging the time-of-day sales patterns to enhance the overall customer experience. Consider adjusting staffing levels, promotional offers, or personalized services during high-average sales periods.
12. Exploration of opportunities for cross-brand promotions during weeks with observed peaks. Capitalize on customer preferences during peak periods to introduce complementary products or cross-brand promotions.
13. Beyond average sales values, evaluate the product preferences and behavior patterns within different age groups. Tailoring marketing messages and promotions to resonate with the unique preferences of each age segment.
14. Identifying potential seasonal trends affecting sales. Planning targeted promotions or introduce seasonal products based on historical sales patterns during specific months or weeks.

----------EOF----------