E-commerce SQL Analysis

Variable	Description
HOUSEHOLD_KEY	Uniquely identifies each household
AGE_DESC	Estimated age range
MARITAL_STATUS_CODE	Marital Status (A - Married, B- Single, U - Unknown)
INCOME_DESC	Household income
HOMEOWNER_DESC	Homeowner, renter, etc.
HH_COMP_DESC	Household composition
HOUSEHOLD_SIZE_DESC	Size of household up to 5+
KID_CATEGORY_DESC	Number of children present up to 3+

Variable	Description
HOUSEHOLD_KEY	Uniquely identifies each household
BASKET_ID	Uniquely identifies a purchase occasion
DAY	Day when transaction occurred
PRODUCT_ID	Uniquely identifies each product
QUANTITY	Number of the products purchased during the trip
SALES_VALUE	Amount of dollars retailer receives from sale
STORE_ID	Identifies unique stores
COUPON_MATCH_DISC	Discount applied due to retailer's match of manufacturer coupon
COUPON_DISC	Discount applied due to manufacturer coupon
RETAIL_DISC	Discount applied due to retailer's loyalty card program
TRANS_TIME	Time of day when the transaction occurred
WEEK_NO	Week of the transaction. Ranges 1 - 102

Variable	Description
PRODUCT_ID	Number that uniquely identifies each product
DEPARTMENT	Groups similar products together
COMMODITY_DESC	Groups similar products together at a lower level
SUB_COMMODITY_DESC	Groups similar products together at the lowest level
MANUFACTURER	Code that links products with same manufacturer together
BRAND	Indicates Private or National label brand
CURR_SIZE_OF_PRODUCT	Indicates package size (not available for all products)

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

```
SELECT SIZE, count(BASKET_ID)

FROM

(SELECT BASKET_ID,

      CASE WHEN SALES_VALUE >= 0 AND SALES_VALUE <=10 THEN 'small'

           WHEN SALES_VALUE >=10 AND SALES_VALUE <=20 THEN 'medium'

           WHEN SALES_VALUE >=20 THEN 'large'

       END AS SIZE

FROM `Sales_Data.transaction_data`) TBL

GROUP BY SIZE;
```

Row	SIZE	TOTAL_ORDERS
1	small	1259081
2	large	12536
3	medium	26869

Insights

1. **Order Distribution**:

   A significant percentage of orders fall into the 'small' category (0-10 dollars), indicating that customers might be purchasing low-value items frequently.

2. **Customer Segmentation**:
   - The data suggests a natural segmentation of customers based on their spending behavior. Customers in the 'small' category might be more

price-sensitive or focused on specific low-cost items, while those in the 'large' category might be less price-sensitive and more likely to engage in higher-value transactions.

Recommendations

- The presence of medium and large orders suggests an opportunity to upsell or cross-sell products. Customers who are already spending in the small and medium category might be encouraged to spend more with the right incentives, such as discounts on bundled products or loyalty rewards for higher spending.

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

```
SELECT SIZE, count(BASKET_ID) AS TOTAL_ORDERS

FROM

(SELECT BASKET_ID,

      CASE WHEN SALES_VALUE >= 0 AND SALES_VALUE <=5 THEN 'small'

           WHEN SALES_VALUE >=5 AND SALES_VALUE <=10 THEN 'medium'

           WHEN SALES_VALUE >=10 THEN 'large'

       END AS SIZE

FROM `Sales_Data.transaction_data`) tbl

GROUP BY SIZE;
```

Row	SIZE	TOTAL_ORDERS
1	small	1145982
2	medium	113099
3	large	39405

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

```
WITH RankedStores AS (
   SELECT
       WEEK_NO,
       STORE_ID,
       COUNT(Distinct BASKET_ID) AS FOOT_TRAFFIC,
       ROW_NUMBER() OVER (PARTITION BY WEEK_NO ORDER BY COUNT(Distinct
BASKET_ID) DESC) AS rank
   FROM
       `Sales_Data.transaction_data`
   GROUP BY
        WEEK_NO, STORE_ID)
SELECT
   WEEK_NO,
   STORE_ID,
   FOOT_TRAFFIC
FROM
   RankedStores
WHERE
   rank <= 3
```

```
ORDER BY
    WEEK_NO;
```

Row	WEEK_NO	STORE_ID	FOOT_TRAFFIC
1	1	32004	8
2	1	296	6
3	1	324	6
4	2	313	13
5	2	292	12
6	2	32004	11
7	3	367	23
8	3	375	22
9	3	32004	15
10	4	367	42

(Ordered by Foot_Traffic- Top-Performing Store is 367)

Row	WEEK_NO	STORE_ID	FOOT_TRAFFIC
1	17	367	108
2	22	367	92
3	65	367	90
4	18	367	89
5	46	367	88
6	83	367	85
7	61	367	83
8	26	361	82
9	63	367	82
10	73	367	80
11	60	367	80

Insights and Recommendations

Top-Performing Stores:

- This query identifies the top three stores with the highest foot traffic each week, providing insight into which locations consistently attract the most customers. These stores likely have strong local demand, effective marketing strategies, or other favorable conditions contributing to their high customer count.

Leverage High-Traffic Stores:

- **Targeted Promotions**: For stores consistently ranking in the top three, consider running special promotions or product launches. Leveraging these high-traffic locations can maximize the impact of marketing efforts and drive higher sales.
- **Stock Optimization**: Ensure that high-traffic stores are always well-stocked with popular products to meet customer demand. Regularly analyze foot traffic data to adjust inventory levels in response to fluctuations in customer visits.
- **Benchmarking**: Compare the operations, customer experience, and marketing efforts of the top-performing stores with lower-ranking ones. Identifying and replicating the success factors from high-traffic stores could help improve performance across the network.

Question 4: Create a basic customer profiling with first, last visit, number of visits, average money spent per visit and total money spent ordered by highest avg money.

```
SELECT household_key,

   MIN(DAY) AS first_visit,

   MAX(DAY) AS last_visit,

   COUNT(DISTINCT BASKET_ID) AS number_of_visits,

   --Number of distinct visits (baskets)

   SUM(SALES_VALUE) / COUNT(DISTINCT BASKET_ID) AS
avg_money_spent_per_visit,  --Average money spent per visit

   SUM(SALES_VALUE) AS total_money_spent

FROM `Sales_Data.transaction_data`
```

```
GROUP BY household_key

ORDER BY avg_money_spent_per_visit desc;
```

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

Row	household_key	first_visit	last_visit	number_of_visits	avg_money_spent_per_visit	total_money_spent
1	2042	52	683	26	89.97	2339.21
2	973	95	710	80	85.95	6875.89
3	1899	20	705	69	83.91	5789.59
4	1900	111	707	55	76.87	4227.72
5	1574	107	651	27	68.27	1843.3
6	1315	60	624	5	63.48	317.39
7	2479	111	706	111	62.65	6954.64
8	931	94	668	40	61.38	2455.29
9	1344	87	691	26	60.4	1570.37
10	248	29	704	53	58.32	3090.89
11	688	70	692	27	57.74	1558.95

Results per page: 50 | 1 – 50 of 2500

Insights

1. **High-Spending Customers**:
    - The query ranks customers by the average money spent per visit. This highlights the most valuable customers in terms of spending behavior, providing a clear picture of who contributes the most revenue on a per-visit basis.
2. **Loyalty and Frequency**:
    - By examining the number of visits, we can identify loyal customers who visit frequently, even if their average spend per visit might be lower. These customers represent consistent revenue streams and may be more receptive to loyalty programs or personalized offers.
3. **Customer Lifetime Value (CLV) Indicators**:
    - The total money spent by each customer combined with their visit frequency and average spend provides **insights into their potential lifetime value**. Customers who have both high total spending and high visit frequency are likely to be among the most valuable for the business.

Recommendations

1. **Enhance Engagement with High-Spending Customers**:

- **Personalized Offers**: Create targeted promotions and personalized offers for customers with the highest average spend per visit. These could include exclusive discounts, early access to sales, or special loyalty rewards tailored to their shopping preferences.

  .

2. **Increase Visit Frequency for High-Spend, Low-Visit Customers**:
   - **Loyalty Programs**: Implement or enhance loyalty programs that reward customers for frequent visits. Points, discounts, or other rewards for each visit can incentivize customers who spend a lot but visit less frequently to increase their shopping frequency.
   - **Re-Engagement Campaigns**: Identify customers who have high average spends but haven't visited recently. Re-engagement campaigns, such as personalized emails or special discounts, can encourage these customers to return.
3. **Cross-Sell and Upsell Opportunities**:
   - **Product Recommendations**: Use the profile data to recommend complementary products or services that align with high-spending customers' preferences. Cross-sell and upsell strategies can increase their average basket size.
   - **Bundled Offers**: Create bundled offers that appeal to high-spending customers, potentially increasing their spend per visit by encouraging them to purchase more items in a single transaction.

Question 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)

```
SELECT demo.*, tbl.total_sales

FROM `Sales_Data.hh_demographic` AS demo

JOIN (

   SELECT
```

```
        household_key,
        ROUND(SUM(SALES_VALUE), 2) AS total_sales
    FROM
        `Sales_Data.transaction_data`
    GROUP BY
        household_key
) AS tbl
ON demo.household_key = tbl.household_key
ORDER BY
    tbl.total_sales DESC
LIMIT 1;
```

Row	AGE_DESC	MARITAL_STATUS_CODE	INCOME_DESC	HOMEOWNER_DESC
1	45-54	A	125-149K	Homeowner

Row	H_COMP_DESC	HOUSEHOLD_SIZE_DESC	KID_CATEGORY_DESC	household_key	total_sales
1	Adults Kids	5+	3+	1609	13804.38

Insights

1. **High-Value Customer Profile**:
    - The query identifies the customer with the highest spending and provides detailed demographic information. This customer's spending habits, combined with their demographic data, make them an ideal candidate for personalized marketing efforts.
2. **Key Demographic Factors**:
    - The demographic data (such as income level, household size, homeownership status, etc.) allows for a deeper understanding of the customer's lifestyle and purchasing power. This can reveal patterns such as high-income households potentially having higher disposable income to spend on premium products.

Recommendations

Expand Demographic Targeting:

- **Segment and Target Similar Customers**: Use the demographic profile of this high-value customer to identify and target similar segments within the customer base. This can be done through targeted advertising to reach potential customers with similar demographics.
- **Marketing Campaigns**: Design marketing campaigns that resonate with the key demographic attributes of this high-spending customer. For example, if the customer is part of a larger household, promotions could emphasize value packs or family-oriented products.

Question 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)

```
WITH CTE AS (

   SELECT t.*, p.*
```

```
    FROM `Sales_Data.transaction_data` AS t
    JOIN `Sales_Data.product` AS p
    ON t.PRODUCT_ID = p.PRODUCT_ID
)

SELECT
    LEAST(c1.SUB_COMMODITY_DESC, c2.SUB_COMMODITY_DESC) AS product_1,
    GREATEST(c1.SUB_COMMODITY_DESC, c2.SUB_COMMODITY_DESC) AS product_2,
    COUNT(*) AS combo_count
FROM
    CTE AS c1
JOIN
    CTE AS c2
ON
    c1.BASKET_ID = c2.BASKET_ID
AND
    c1.SUB_COMMODITY_DESC < c2.SUB_COMMODITY_DESC
GROUP BY
    LEAST(c1.SUB_COMMODITY_DESC, c2.SUB_COMMODITY_DESC),
    GREATEST(c1.SUB_COMMODITY_DESC, c2.SUB_COMMODITY_DESC)
```

```
ORDER BY
    combo_count DESC;
```

Row	product_1	product_2	combo_count
1	FLUID MILK WHITE ONLY	YOGURT NOT MULTI-PACKS	5953
2	BANANAS	FLUID MILK WHITE ONLY	4365
3	FLUID MILK WHITE ONLY	SOFT DRINKS 12/18&15PK CA...	4326
4	FLUID MILK WHITE ONLY	MAINSTREAM WHITE BREAD	3934
5	BANANAS	YOGURT NOT MULTI-PACKS	3847
6	FLUID MILK WHITE ONLY	SHREDDED CHEESE	3840
7	FLUID MILK WHITE ONLY	SFT DRNK 2 LITER BTL CARB I...	3494
8	FRZN SS PREMIUM ENTREES/...	YOGURT NOT MULTI-PACKS	3344
9	BABY FOOD - BEGINNER	BABY FOOD JUNIOR ALL BRAN...	3290

Insights and Recommendations

1. **Frequently Bought Together Products**:
    - The query identifies pairs of products that are most frequently purchased together by customers, and the count of each combination. This can reveal common customer behaviors, such as a tendency to purchase certain items in pairs, which can be leveraged for strategic decisions.
2. **Inventory Management**:
    - Understanding which products are often bought together can help optimize inventory management. Ensuring that these commonly paired items are stocked together or near each other in stores could enhance the customer shopping experience.
3. **Promotion Strategies**:
    - The most frequent product combinations can be the focus of targeted promotions or bundled discounts. For instance, offering a discount when purchasing "Product A" and "Product B" together could increase sales and enhance customer satisfaction.

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

```
   WITH CTE AS (
   SELECT
       household_key,
       WEEK_NO,
       COUNT(*) AS count,
       ROUND(SUM(sales_value), 2) AS revenue_per_week
   FROM
       `Sales_Data.transaction_data`
   GROUP BY
       household_key, WEEK_NO
   ORDER BY
       household_key, WEEK_NO
)


SELECT
   *,
   ROUND(revenue_per_week - last_week_revenue, 2) AS diff
FROM (
```

```
    SELECT
        CTE.*,
        LAG(cte.revenue_per_week) OVER(ORDER BY household_key, WEEK_NO)
AS last_week_revenue
    FROM
        CTE
) tbl
ORDER BY
    household_key, WEEK_NO;
```

Row	household_key	WEEK_NO	count	revenue_per_week	last_week_revenue	diff
1	1	8	14	42.58	*null*	*null*
2	1	10	5	14.01	42.58	-28.57
3	1	13	6	14.03	14.01	0.02
4	1	14	9	25.71	14.03	11.68
5	1	15	5	10.98	25.71	-14.73
6	1	16	4	9.09	10.98	-1.89
7	1	17	5	13.98	9.09	4.89
8	1	19	14	47.35	13.98	33.37
9	1	20	14	31.77	47.35	-15.58
10	1	22	16	38.98	31.77	7.21
11	1	23	9	26.36	38.98	-12.62

Insights

1. **Revenue Per Account (RPA) Trends**:
    - The query calculates the weekly change in Revenue Per Account (RPA) for each customer, comparing the revenue generated each week with the previous week. This analysis can reveal patterns in customer spending behavior over time, such as consistent growth, decline, or volatility.
2. **Customer Segmentation**:
    - By observing the weekly changes in RPA, you can segment customers into different categories based on their spending patterns. For example, customers with consistent RPA growth could be targeted with premium

offerings, while those with declining RPA may need re-engagement strategies.

3. **Customer Retention**:
    - A decline in RPA over multiple weeks might suggest a risk of churn. Monitoring these trends allows for early identification of customers who might be reducing their engagement with the brand, providing an opportunity to re-engage them before they churn.

Recommendations

1. **Re-engagement Strategies**:
    - **Early Churn Indicators**: Develop a re-engagement strategy for customers who exhibit a decline in RPA over consecutive weeks. This could involve personalized communication, special offers, or a customer satisfaction survey to understand and address their concerns.
    - **Retention Campaigns**: Create retention campaigns specifically targeting segments identified as at risk of reducing their spending or churning. Highlight value propositions or remind them of the benefits they enjoy with your products.
2. **Campaign Performance Analysis**:
    - **Monitor Post-Campaign RPA**: After launching marketing campaigns, closely monitor the changes in RPA to assess the campaign's impact. If a positive change is observed, replicate successful elements in future campaigns; if negative, analyze what went wrong and adjust accordingly.
    - **A/B Testing**: Use the RPA data to conduct A/B testing of different marketing strategies. Compare the RPA changes between different customer segments exposed to various marketing tactics to identify the most effective approach.

Question 8: Find retained customers quarter over quarter

```
WITH CTE AS (

  SELECT *,

  CASE

      WHEN DAY >= 1 AND DAY <= 90 THEN 1
```

```
      WHEN DAY > 90 AND DAY <= 180 THEN 2
      WHEN DAY > 180 AND DAY <= 270 THEN 3
      WHEN DAY > 270 AND DAY <= 360 THEN 4
      WHEN DAY > 360 AND DAY <= 450 THEN 5
      WHEN DAY > 450 AND DAY <= 540 THEN 6
      WHEN DAY > 540 AND DAY <= 630 THEN 7
      WHEN DAY > 630 AND DAY <= 711 THEN 8
  END as Quarter
  FROM `Sales_Data.transaction_data`
)



-- Find the customer retention quarter over quarter
SELECT
   this_quarter.Quarter,
COUNT(DISTINCT this_quarter.household_key) AS
total_customers_this_quarter,
COUNT(DISTINCT last_quarter.household_key) AS retained_customers,
ROUND(COUNT(DISTINCT last_quarter.household_key) * 100.0 /
COUNT(DISTINCT this_quarter.household_key), 2) AS retention_rate
FROM
```

```
  CTE AS this_quarter
LEFT JOIN
  CTE AS last_quarter
ON
  this_quarter.household_key = last_quarter.household_key
  AND this_quarter.Quarter - last_quarter.Quarter = 1
GROUP BY
  this_quarter.Quarter
ORDER BY
  this_quarter.Quarter;
```

Row	Quarter	total_customers_this	retained_customers	retention_rate
1	1	1733	0	0.0
2	2	2352	1601	68.07
3	3	2264	2191	96.78
4	4	2261	2144	94.83
5	5	2275	2146	94.33
6	6	2295	2174	94.73
7	7	2304	2193	95.18
8	8	2291	2175	94.94

Insights and Recommendations

1. **Quarterly Retention Trends**:
    - The query calculates the number of retained customers quarter over quarter, giving a clear picture of customer retention trends. A high retention rate indicates strong customer loyalty and satisfaction, while a declining retention rate might suggest potential issues with customer engagement or satisfaction.

- Decline might correlate with specific external factors (e.g., seasonality, economic conditions) or internal issues (e.g., product changes, service disruptions).
- If the drop-off correlates with specific events or changes in the business, such as a price increase or a product update, consider adjusting those factors.

Question 9: Find customer churn quarter over quarter

```
WITH CTE AS (
  SELECT *,
  CASE
      WHEN DAY >= 1 AND DAY <= 90 THEN 1
      WHEN DAY > 90 AND DAY <= 180 THEN 2
      WHEN DAY > 180 AND DAY <= 270 THEN 3
      WHEN DAY > 270 AND DAY <= 360 THEN 4
      WHEN DAY > 360 AND DAY <= 450 THEN 5
      WHEN DAY > 450 AND DAY <= 540 THEN 6
      WHEN DAY > 540 AND DAY <= 630 THEN 7
      WHEN DAY > 630 AND DAY <= 711 THEN 8
  END AS Quarter
  FROM `Sales_Data.transaction_data`
)
SELECT
```

```
  last_quarter.Quarter,
  COUNT(DISTINCT last_quarter.household_key) AS churned_customers
FROM
  CTE AS last_quarter
LEFT JOIN
  CTE AS this_quarter
ON
  this_quarter.household_key = last_quarter.household_key
  AND this_quarter.Quarter - last_quarter.Quarter = 1
WHERE
  this_quarter.household_key IS NULL
GROUP BY
  last_quarter.Quarter
ORDER BY
  last_quarter.Quarter;
```

Row	Quarter	churned_customers
1	1	132
2	2	161
3	3	120
4	4	115
5	5	101
6	6	102
7	7	129
8	8	2291

Insights and Recommendations

Customer Churn Analysis:

- The query calculates the number of customers churned after each quarter, giving an understanding of how many customers are lost compared to the previous quarter. This information is crucial in understanding customer retention trends and identifying periods where churn might be particularly high.
- Consider segmenting the customer base by demographic or behavioral factors to understand which customer segments have higher churn rates. This can lead to more targeted interventions for at-risk customer groups.

Question 10: During what time of the day householders 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

```
WITH CTE AS(

   SELECT BASKET_ID, TRANS_TIME/100 as formatted_time

   FROM `Sales_Data.transaction_data`

)



SELECT time_of_day, COUNT(DISTINCT BASKET_ID) as orders_count
```

```sql
FROM

(

   SELECT *,

   CASE

        WHEN formatted_time >= 0 AND formatted_time < 7  THEN 'Dawn'

        WHEN formatted_time >= 7 AND formatted_time < 13 THEN 'Morning'

        WHEN formatted_time >= 13 AND formatted_time < 18 THEN
'Afternoon'

        WHEN formatted_time >= 18 AND formatted_time < 24 THEN 'Night'

   END as time_of_day

   FROM CTE

) AS tbl

GROUP BY time_of_day

ORDER BY orders_count DESC;
```

Row	time_of_day	orders_count
1	Afternoon	96507
2	Night	77361
3	Morning	53473
4	Dawn	6015

Insights and recommendations

Order Distribution:

- Afternoons and Nights have the highest number of orders. Businesses could use this insight to offer promotions during slower times to balance the load.
- Understanding when most orders are placed can help optimize operational processes, such as delivery scheduling or customer support availability.
- Segment customers by demographic or purchasing behavior to see if certain groups prefer placing orders at different times.
- Consider running A/B tests with promotions during different times of the day to see if it influences order timing and increases overall sales.