E-commerce SQL Analysis

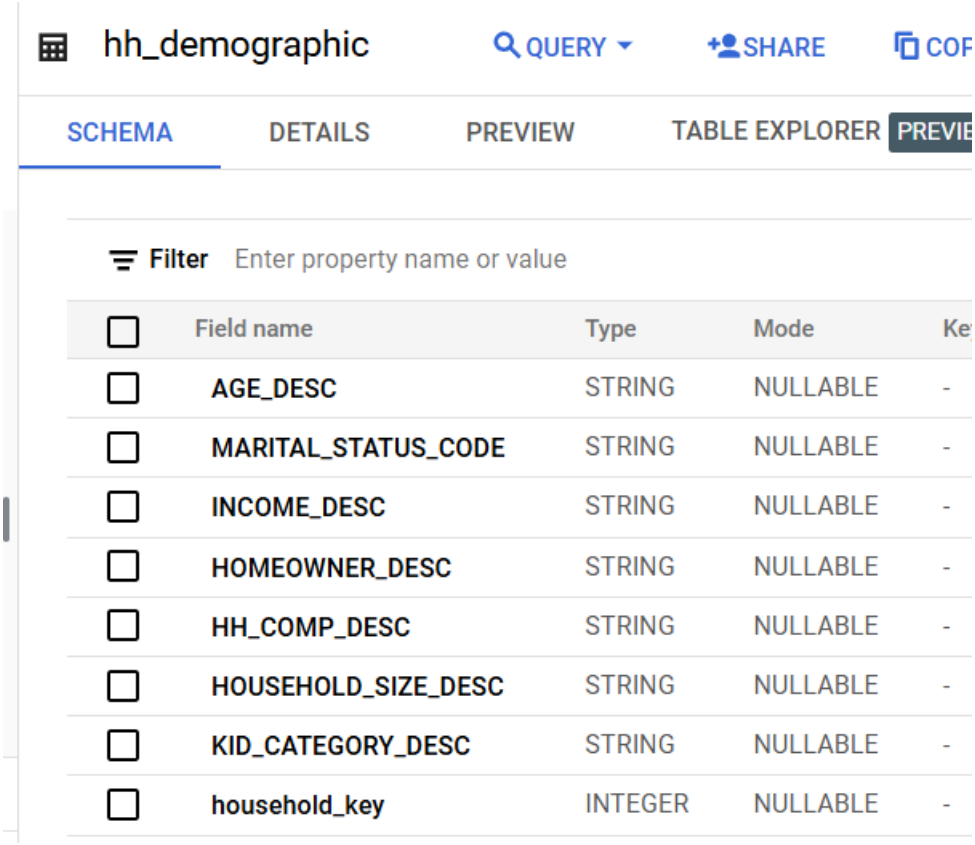
Problem Statement

Analyzing the sales, product, and customer data for an e-commerce company getting various insights and calculating various KPI and data with SQL in Big Query.

Data Dictionary and Schema:

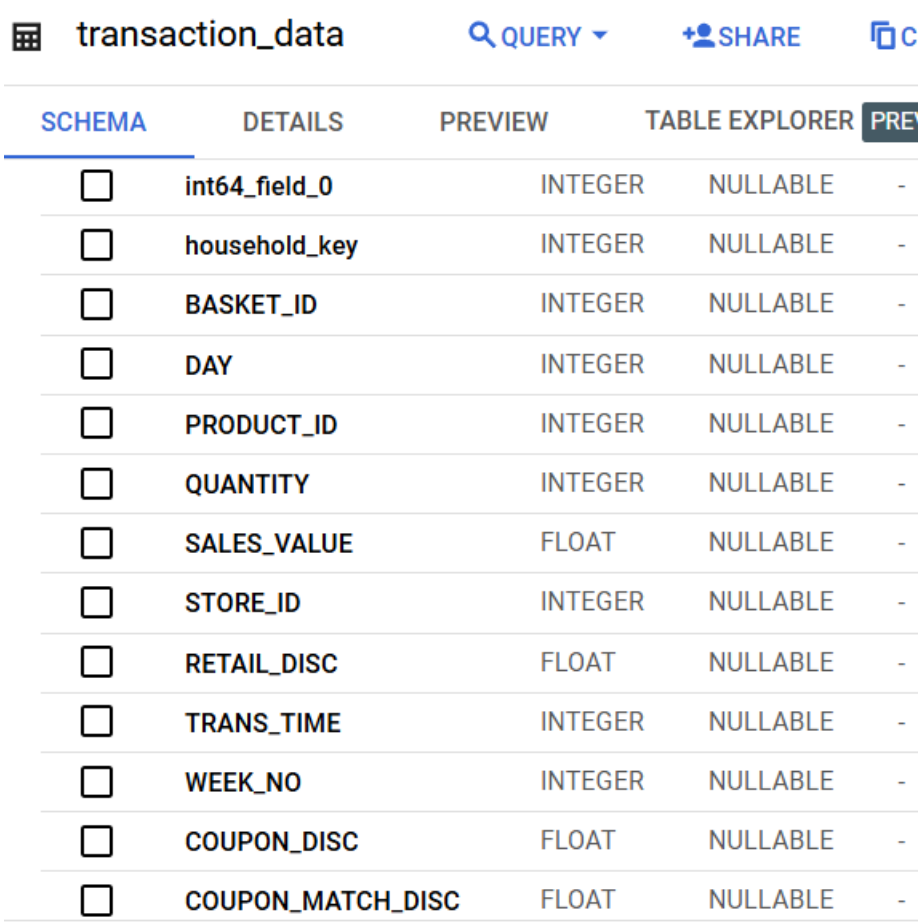
Demographic table

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+



Transaction table:

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



Products Table:

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)

product QUERY SHARE COPY SNAI

SCHEMA DETAILS PREVIEW TABLE EXPLORER PREVIEW

Filter Enter property name or value

Field name	Type	Mode	Key
PRODUCT_ID	INTEGER	NULLABLE	-
MANUFACTURER	INTEGER	NULLABLE	-
DEPARTMENT	STRING	NULLABLE	-
BRAND	STRING	NULLABLE	-
COMMODITY_DESC	STRING	NULLABLE	-
SUB_COMMODITY_DESC	STRING	NULLABLE	-
CURR_SIZE_OF_PRODUCT	STRING	NULLABLE	-

KPI:

1. **Total Sales:**

```
1  select sum(SALES_VALUE) as Total_Sales
2  from `Ecommerce.transaction_data`
```

Row	Total_Sales
1	4029338.410022...

Inference: Total Sales =$4029338

2. Total Transactions:

```
select count(distinct(BASKET_ID)) as Total_transactions
from `Ecommerce.transaction_data`
```

Row	Total_transactions
1	233356

Inference: Total number of transactions=233356

3.Average Order Value:

```sql
with cte as
(select sum(SALES_VALUE) as Total_Sales,count(distinct(BASKET_ID)) as Total_transactions
from `Ecommerce.transaction_data`)

select cte.Total_Sales/cte.Total_transactions as Average_order_value
from cte
```

JOB INFORMATION RE

Row	Average_order_value
1	17.26691582827…

Average Order Value=17.2 $

4.Total Number of Households:

```sql
select count(distinct(household_key)) as Total_households
from `Ecommerce.transaction_data`
```

JOB INFORMATION RE

Row	Total_households
1	2500

Inference: Total Number of households= 2500

5.Total Revenue

```sql
with cte2 as
(select sum(SALES_VALUE) as Total_sales,sum(RETAIL_DISC) as Retail_Discount,sum(COUPON_DISC) as Coupon_Discount,sum(COUPON_MATCH_DISC) as
Coupon_Match_Discount
from `Ecommerce.transaction_data`)

select cte2.Total_Sales+cte2.Retail_Discount+cte2.Coupon_Discount+cte2.Coupon_Match_Discount as Total_Revenue
from cte2
```

Query results

JOB INFORMATION RESULTS

Row	Total_Revenue
1	3304975.560023…

Inference: Total Revenue= 3304975 $

Analysis:

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+)

Answer:

```
#Find the number of orders that have small, medium or large order value (small:0-10 dollars, medium:10-20 dollars, large:20+)
with cte3 as
(select *,sum(SALES_VALUE) over(partition by BASKET_ID) as Total_order_value
from `Ecommerce.transaction_data`)

,cte4 as
(select *,
case
  when 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"
end as Type_of_order

from cte3)

select type_of_order,count(distinct(BASKET_ID)) as count_of_order
from cte4
group by type_of_order
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	type_of_order	count_of_order
1	large	67309
2	small	115045
3	medium	51002

Inference:

There 67309 large(20+$) value orders,51002(10-20$) orders and 115045 small(less than 10$) orders. The majority of orders are of small value.

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+)

Answer:

```sql
#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+)
with cte3 as
(select *,sum(SALES_VALUE) over(partition by BASKET_ID) as Total_order_value
from `Ecommerce.transaction_data`)

,cte5 as
(select *,
case
  when Total_order_value<5 then "small"
  when Total_order_value>=5 and Total_order_value<=10 then "medium"
  when Total_order_value>10 then "large"
end as Type_of_order

from cte3)

select type_of_order,count(distinct(BASKET_ID)) as count_of_order
from cte5
group by type_of_order
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	type_of_order	count_of_order
1	medium	46665
2	large	116939
3	small	69752

Inference:

There 116939 large(10+$) value orders,46665(5-10$) orders and 69752small(less than 5$) orders. The majority of orders are of large value.

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

```sql
#Question 3: Find top 3 stores with highest foot traffic for each week (Foot traffic: number of custo
with cte6 as
(select STORE_ID,WEEK_NO,count(distinct(household_key)) as number_of_customer
from `Ecommerce.transaction_data`
group by STORE_ID,WEEK_NO
order by 2 asc,3 desc)

,cte7 as
(select *,dense_rank() over(partition by WEEK_NO order by cte6.number_of_customer desc) as rnk
from cte6)

select * from
cte7
where rnk<=3
```

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DET

ow	STORE_ID	WEEK_NO	number_of_custome	rnk
22	32004	3	9	2
23	356	3	8	3
24	367	4	17	1
25	32004	4	11	2
26	446	4	8	3
27	367	5	15	1
28	32004	5	13	2
29	292	5	8	3
30	361	5	8	3
31	32004	6	21	1

Inference:

The various stores with high sales each week can be seen. Here store number 367 and 32004 have consistently been having high sales.

Question 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

Answer:

```
#Question 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

select distinct(d.household_key),min(day) over (partition by d.household_key) as first_visit,max(day) over (partition by d.household_key) as last_visit,count(distinct(BASKET_ID)) over(partition by d.household_key) as number_of_visit,sum(SALES_VALUE) over(partition by d.household_key) as total_spent,(sum(SALES_VALUE) over(partition by d.household_key))/(count(distinct(BASKET_ID)) over(partition by d.household_key)) as Ave_money_per_visit
 from
`Ecommerce.hh_demographic` d
left join `Ecommerce.transaction_data` t
on d.household_key=t.household_key
order by 6 desc
```

Query results

SAVE

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

Row	household_key	first_visit	last_visit	number_of_visit	total_spent	Ave_money_per_visit
1	973	95	710	80	6875.89	85.948625
2	1899	20	705	69	5789.59	83.90710144927...
3	2479	111	706	111	6954.64	62.65441441441...
4	248	29	704	53	3090.89	58.31867924528...
5	688	70	692	27	1558.95	57.73888888888...
6	1864	103	710	148	8537.28	57.68432432432...
7	1848	105	706	97	5561.56	57.33567010309...
8	888	12	706	76	4299.36	56.57052631578...
9	392	75	709	103	5657.72	54.92932038834...
10	1662	85	702	79	4329.22	54.80025316455...

Inference:

Customer with household key number 973 has the highest average money spent per visit with 85.9$ even though the number of visits is an average value.

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 household_key,sum(SALES_VALUE) as total_spend
from `Ecommerce.transaction_data`
group by 1
order by 2 desc
```

Query results

JOB INFORMATION | RESULTS | CHART

Row	household_key	total_spend
1	1023	18901.09000000...
2	1609	13804.37999999...
3	2322	11934.65999999...
4	1453	10720.71999999...

From above we can see customer with household_key=1023 has highest total spend of $18901. But there is no information about this customer in demographic table.

So we take the next highest customer with household_key=1609

```
select * from
`Ecommerce.hh_demographic` d
inner join `Ecommerce.transaction_data` t
on d.household_key=t.household_key
where t.household_key=1609
```

Query results

SAVE RESULTS EXPLORE DATA

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

Row	HOMEOWNER_DESC	HH_COMP_DESC	HOUSEHOLD_SIZE_DESC	KID_CATEGORY_DESC	household_key	int64_field_0
1	Homeowner	2 Adults Kids	5+	3+	1609	3638
2	Homeowner	2 Adults Kids	5+	3+	1609	3440
3	Homeowner	2 Adults Kids	5+	3+	1609	3464
4	Homeowner	2 Adults Kids	5+	3+	1609	3496
5	Homeowner	2 Adults Kids	5+	3+	1609	3683
6	Homeowner	2 Adults Kids	5+	3+	1609	4073
7	Homeowner	2 Adults Kids	5+	3+	1609	4456
8	Homeowner	2 Adults Kids	5+	3+	1609	5265
9	Homeowner	2 Adults Kids	5+	3+	1609	5498
10	Homeowner	2 Adults Kids	5+	3+	1609	5766

Inference:

The most spending customer with other demographic details is customer with household key value 1609. The customer is in the age group of 45-54 and has high income of 125-149K $.They are married and have 3 kids.

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)

```
#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 cte8 as
(select t.PRODUCT_ID,p.COMMODITY_DESC,p.SUB_COMMODITY_DESC,t.BASKET_ID
from `Ecommerce.product` p
inner join `Ecommerce.transaction_data` t
on p.PRODUCT_ID=t.PRODUCT_ID)


select t1.SUB_COMMODITY_DESC, t2.SUB_COMMODITY_DESC, count(distinct t1.BASKET_ID) as count_orders
from cte8 t1 join
    cte8 t2
    on t1.BASKET_ID = t2.BASKET_ID and
      t1.SUB_COMMODITY_DESC < t2.SUB_COMMODITY_DESC
group by t1.SUB_COMMODITY_DESC, t2.SUB_COMMODITY_DESC
order by count_orders desc;
```

Query results

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

Row	SUB_COMMODITY_DESC	SUB_COMMODITY_DESC_1	count_orders
1	BANANAS	FLUID MILK WHITE ONLY	4131
2	FLUID MILK WHITE ONLY	MAINSTREAM WHITE BREAD	3753
3	FLUID MILK WHITE ONLY	SOFT DRINKS 12/18&15PK CA...	3328
4	FLUID MILK WHITE ONLY	SHREDDED CHEESE	3155
5	FLUID MILK WHITE ONLY	YOGURT NOT MULTI-PACKS	2805
6	DAIRY CASE 100% PURE JUICE...	FLUID MILK WHITE ONLY	2682
7	FLUID MILK WHITE ONLY	SFT DRNK 2 LITER BTL CARB I...	2579
8	FLUID MILK WHITE ONLY	KIDS CEREAL	2554
9	FLUID MILK WHITE ONLY	POTATO CHIPS	2200
10	EGGS - LARGE	FLUID MILK WHITE ONLY	1952
11	BANANAS	YOGURT NOT MULTI-PACKS	1931

Inference:

The combination of Bananas with Fluid milk white is the highest selling combination with it being bought together 4131 times by customers.

Question 7:

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

```sql
#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 cte9 as
(select household_key,week_no,sum(SALES_VALUE) as current_week_spend
from `Ecommerce.transaction_data`
group by household_key,WEEK_NO
order by 1,2)

,cte10 as
(select *,lag(cte9.current_week_spend) over(partition by household_key order by week_no) as prev_week_spend
from cte9
order by household_key,week_no)

select *,current_week_spend-prev_week_spend as Spend_diff
from cte10
```

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

Row	household_key	week_no	current_week_spend	prev_week_spend	Spend_diff
1	1	8	42.58	*null*	*null*
2	1	10	14.01	42.58	-28.57
3	1	13	14.03000000000…	14.01	0.020000000000…
4	1	14	25.71	14.03000000000…	11.68
5	1	15	10.98	25.71	-14.73
6	1	16	9.09	10.98	-1.89000000000…
7	1	17	13.98	9.09	4.890000000000…
8	1	19	47.35000000000…	13.98	33.37000000000…
9	1	20	31.77	47.35000000000…	-15.5800000000…
10	1	22	38.98000000000…	31.77	7.210000000000…
11	1	23	26.36	38.98000000000…	-12.6200000000…

Inference:

From the above we cans see that the spendings keep varying with each week while some weeks have high spends most of the other weeks there are lower amount spend.

Question 8:

Which week has maximum sale?

```sql
#Q 8:Which week has maximum sales?
select week_no,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data`
group by WEEK_NO
order by 2 desc
```

Query results

JOB INFORMATION | RESULTS | CHART

Row	week_no	Total_sales
1	92	57721.89999999...
2	99	50361.01999999...
3	85	49506.66999999...
4	98	49447.69999999...
5	94	49079.32999999...
6	68	48781.50999999...
7	72	48733.54999999...
8	46	48189.51999999...
9	59	48115.44999999...
10	90	46635.18999999...

Inference:

The weeks 92,99,98,94 etc as shown above has high sales. Thus company needs to ensure adequate inventory is available on those weeks.

Question 9:

Which are the top 10 products bought by consumers.

```
#Q 9:Which product is bought maximum?

select p.sub_commodity_desc,sum(SALES_VALUE) as Total_Sales from
`Ecommerce.transaction_data` t
inner join
`Ecommerce.product`p
on t.PRODUCT_ID=p.product_id
group by p.sub_commodity_desc
order by 2 desc
```

JOB INFORMATION | RESULTS | CHART | JSON | EXEC

Row	sub_commodity_desc	Total_Sales
1	GASOLINE-REG UNLEADED	315997.0900000...
2	FLUID MILK WHITE ONLY	80754.44000000...
3	SOFT DRINKS 12/18&15PK CA...	79214.43999999...
4	BEERALEMALT LIQUORS	75036.18000000...
5	CIGARETTES	48179.15000000...
6	CHOICE BEEF	38382.08999999...
7	SHREDDED CHEESE	34252.77999999...
8	PRIMAL	32829.33000000...
9	PREMIUM	32537.04000000...
10	BABY DIAPERS	30421.63000000...
11	TOILET TISSUE	29433.64000000...

Inference:

The product that is sold the most is Gasoline -reg unleaded.

Question 10:

Which are the top 10 department and which product category contributes most to it.

```
select p.department,sum(SALES_VALUE) as Total_Sales
from
`Ecommerce.transaction_data` t
inner join
`Ecommerce.product`p
on t.PRODUCT_ID=p.product_id
group by p.department
order by 2 desc
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DET

ow	department	Total_Sales
1	GROCERY	2046695.129995...
2	DRUG GM	527588.6499997...
3	PRODUCE	279720.3899998...
4	MEAT	274036.3199999...
5	KIOSK-GAS	269461.6700000...
6	MEAT-PCKGD	206491.7099999...
7	DELI	130322.2600000...
8	MISC SALES TRAN	62633.94999999...
9	PASTRY	61786.55999999...
10	NUTRITION	48840.38999999...

Inference:

The department with high sales are Grocery,Drug,Produce,Meat etc.

```
with cte11 as
(select p.department,p.commodity_desc,sum(SALES_VALUE) as Total_Sales
from
`Ecommerce.transaction_data` t
inner join
`Ecommerce.product`p
on t.PRODUCT_ID=p.product_id
group by p.department,p.commodity_desc
order by 3 desc)

,cte12 as
(select department,commodity_desc,Total_sales,dense_rank() over (partition by department order by cte11.Total_Sales desc) as rnk
from cte11)

select department,commodity_desc,Total_sales
from cte12
where rnk=1
order by Total_sales desc
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

Row	department	commodity_desc	Total_sales
1	KIOSK-GAS	COUPON/MISC ITEMS	254675.6000000...
2	GROCERY	SOFT DRINKS	164139.5999999...
3	MEAT	BEEF	156390.7799999...
4	MISC SALES TRAN	COUPON/MISC ITEMS	62633.94999999...
5	DELI	DELI MEATS	52911.36999999...
6	DRUG GM	CIGARETTES	48176.29000000...
7	MEAT-PCKGD	LUNCHMEAT	46214.30999999...
8	SEAFOOD-PCKGD	SEAFOOD - FROZEN	25285.80000000...
9	PRODUCE	POTATOES	22683.46000000...
10	PASTRY	CAKES	19836.69000000...
11	SALAD BAR	SALAD BAR	15709.85000000...

Inference:

The product category with highest sales are seen above under respective departments.

Question 11:

Which are the top 10 store that has the highest sale?

```
# Q 11:Which store has the highest sales?
select store_id,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data`
group by STORE_ID
order by 2 desc
```

JOB INFORMATION | RESULTS | CHART | JSO

Row	store_id	Total_sales
1	367	134105.4600000...
2	406	108814.6600000...
3	361	72493.71999999...
4	429	70752.60999999...
5	343	70265.59999999...
6	356	69026.01999999...
7	375	65788.49999999...
8	381	65400.57999999...
9	292	65201.59999999...
10	31782	61012.22999999...
11	321	59675.479999998

Inference:

The stores with high sales are shown above.

Question 12:

Which stores have the lowest sales?

```
# Q 12:Which stores has the lowest sales?
select store_id,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data`
group by STORE_ID
order by 2
```

Query results

JOB INFORMATION | RESULTS | CHART | JS

Row	store_id	Total_sales
1	1235	0.0
2	3932	0.25
3	681	0.4
4	564	0.41
5	256	0.76
6	2825	0.870000000000...
7	3073	0.95
8	765	1.0
9	2760	1.0
10	2743	1.0

Inference:

The stores with low sales are shown above.Need to check on why these stores have least sales. Whether they are recently opened or not. If they are older stores then why the sale has been low. Then management needs to take decision on whether to add promotions or close the store.

Question 13:

Analyze Spend by Marital Status.

```
# Sales by Marital Status
select marital_status_code,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data` t
inner join `Ecommerce.hh_demographic` d
on t.household_key=d.household_key
group by 1
order by 2 desc
```

Query results

JOB INFORMATION | RESULTS | CHART | JSO

ow	marital_status_code	Total_sales
1	A	1045561.929999...
2	U	904703.9299995...
3	B	299810.9899999...

Inference :

The married people are the ones with the highest spend.

Question 14:

Analyze Spend by Age.

```
# Sales by Age group
select AGE_DESC,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data` t
inner join `Ecommerce.hh_demographic` d
on t.household_key=d.household_key
group by 1
order by 2 desc
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

Row	AGE_DESC	Total_sales
1	45-54	827984.8999995...
2	35-44	622164.3499997...
3	25-34	389545.1699998...
4	65+	151606.8100000...
5	55-64	150371.2700000...
6	19-24	108404.3500000...

Inference:

The age group 45-54 has the highest sales since they are the ones with higher income.

Question 14:

Analyze Spend by Homeowner Description.

```
# Sales by Homeowner Desc
select Homeowner_DESC,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data` t
inner join `Ecommerce.hh_demographic` d
on t.household_key=d.household_key
group by 1
order by 2 desc
```

JOB INFORMATION | RESULTS | CHART | JSO

Row	Homeowner_DESC	Total_sales
1	Homeowner	1519166.589998...
2	Unknown	561266.7799997...
3	Renter	118735.7700000...
4	Probable Owner	27236.49000000...
5	Probable Renter	23671.22000000...

Inference:

The HomeOwners have the highest purchase amount.

Question 15:

Analyze Spend by Income

```
# Sales by Income Desc
select Income_DESC,sum(sales_value) as Total_sales
from `Ecommerce.transaction_data` t
inner join `Ecommerce.hh_demographic` d
on t.household_key=d.household_key
group by 1
order by 2 desc
```

JOB INFORMATION | RESULTS | CHART | JSON | EXEC

Row	Income_DESC	Total_sales
1	50-74K	547139.0499997...
2	35-49K	414471.7899999...
3	75-99K	279738.2199999...
4	25-34K	189846.9300000...
5	Under 15K	169160.1900000...
6	15-24K	151340.2800000...
7	125-149K	150464.7000000...
8	150-174K	126501.6900000...
9	100-124K	100931.4700000...

Load more

Inference:

The people in the income range of 50-74k have the highest purchasing power.

Question 16:

Analyse sales by time of Day:

```
#Sale by Time of day
with cte14 as
(select *,TRANS_TIME/100 as timeofpurchase
from `Ecommerce.transaction_data`)

,cte15 as
(select basket_id,SALES_VALUE,
case
  when timeofpurchase>23 and timeofpurchase<=3 then "midnight"
  when timeofpurchase>3 and timeofpurchase<=6 then "early morning"
  when timeofpurchase>6 and timeofpurchase<=11 then "morning"
  when timeofpurchase>11 and timeofpurchase<=15 then "afternoon"
  when timeofpurchase>15 and timeofpurchase<=19 then "evening"
  when timeofpurchase>19 and timeofpurchase<=23 then "night"
end as Day_time_split
from cte14)

select Day_time_split,sum(sales_value)
from cte15
group by 1
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON

ow	Day_time_split	f0_
1	evening	1506469.399998...
2	afternoon	1181713.119999...
3	night	834832.8999995...
4	morning	403459.6599999...
5	*null*	86651.39000000...
6	early morning	16211.93999999...

Inference:

Maximum purchases are made in the evening from 3pm-7pm.

Recommendations:

.

- **Inventory Management of Popular Product Categories:** A large share of sales comes from five key departments: grocery, drugGM, KIOSK Gas, produce, and meat. Ensure adequate inventory of these items is maintained. Promoting these popular areas and offering appealing deals will encourage more purchases, while a broad selection and competitive pricing will meet customer demand and drive overall sales growth.
- **Targeted Marketing Strategies:** The majority of transactions are driven by homeowners, especially those aged 45–54 with an income between $35K and $74K. Most purchases occur in the evening. Marketing efforts should be customized to cater to this demographic. This could include creating tailored promotions, product selections, and advertising campaigns that resonate with their preferences and needs
- **Cater to Married Customers:** Since married customers generate the majority of total revenue, it's important to focus on meeting their needs and preferences. Understanding their motivations and providing personalized offers can enhance their shopping experience, leading to increased loyalty and higher sales.
- **Bundled offers:** Considering that the highest selling product is Gasoline-unleaded the company can introduce offers for other items along with gasoline so that people would be attracted to buy them.Also looking at the most best selling combos like banana and milk or bread and milk ,then maybe modify to add another element to this combo to encourage higher purchase.
- **Placement of products:** The company can arrange the online portal based on products that sell together so that they are visible near each other. For example soft drink along with milk etc.
- **Maximize Revenue from Key Age Groups:** With most transactions coming from customers aged 45–54, it's essential to concentrate marketing efforts on this age

group. Offering products and services that align with their interests can help maximize revenue from this demographic.

- **Offer Personalized Deals for High Spenders:** By identifying high-spending customers, such as homeowners or those in specific age groups, you can implement targeted marketing and personalized offers. Providing these customers with exclusive discounts, loyalty rewards, or special privileges can encourage repeat business and boost customer satisfaction.
- **Introduce a Loyalty Program:** A loyalty program that offers incentives, rewards, and exclusive deals can help drive repeat purchases and improve customer retention. Customizing loyalty programs for homeowners, married customers, and specific age groups can further enhance their effectiveness.