Project by Faraz

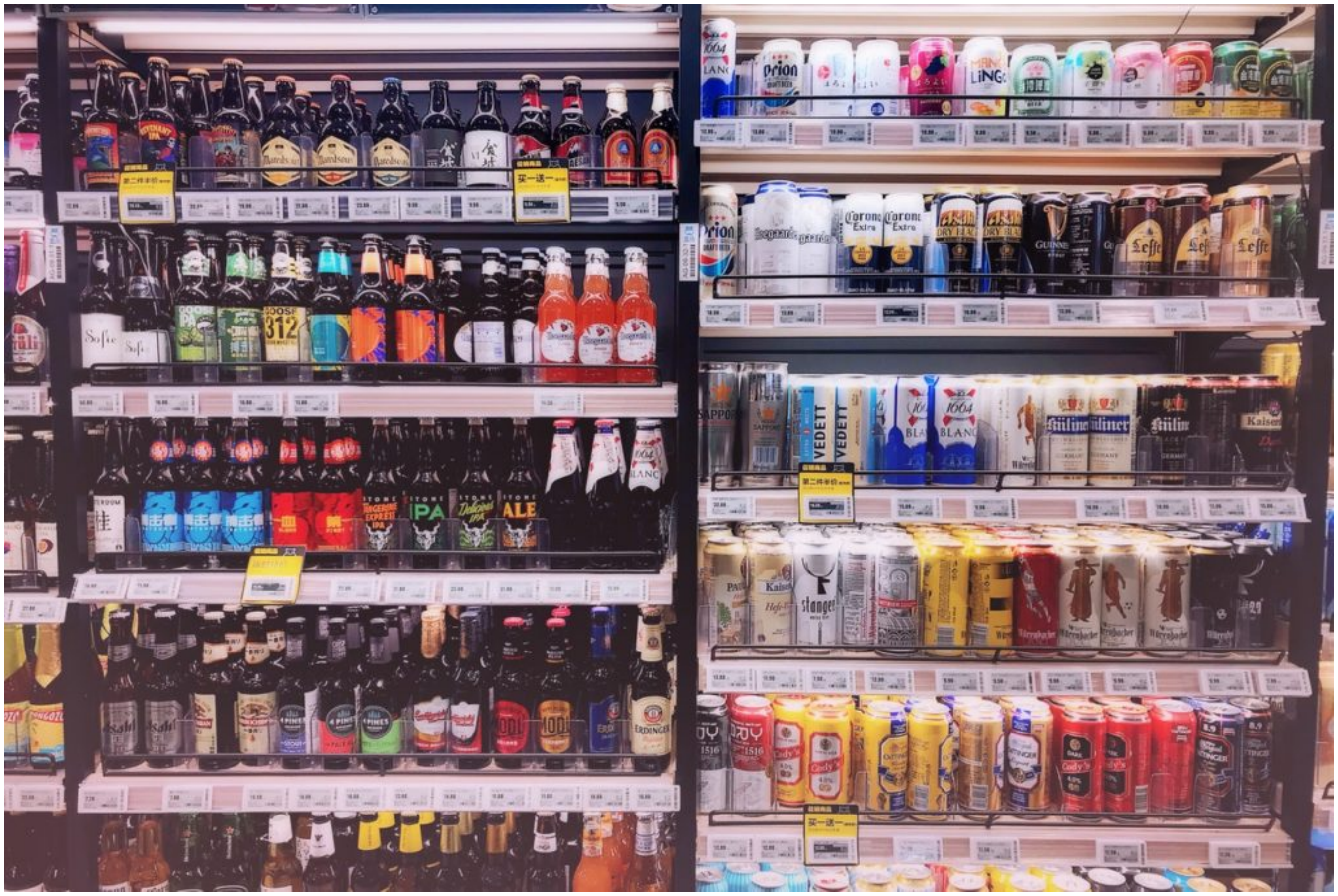


Table of Content

1. Introduction:

This report aims to provide valuable insights into the performance of a soda distribution company's product portfolio. The primary focus will be on analyzing the cost of marketing and marketing-to-sales ratio for different products and product classes, with a specific emphasis on caffeinated and non-caffeinated beverages. By comparing these ratios, we will identify top-performing products with higher sales and lower marketing investment, as well as underperforming products that may require further investigation.

In addition, we will address key business questions such as which product generates the most sales, the marketing-to-sales ratio for various products, the comparison of caffeinated and non-caffeinated beverage sales, and the gross profit margin for different products. The findings from this analysis will enable the

Loading [MathJax]/extensions/Safe.js

company to make informed decisions about prioritizing product categories, refining marketing strategies, and identifying growth opportunities.

2. Business Objective(s):

- Analyzing the cost of marketing and marketing to sales ratio of different product and product classes.
- Identifying top performers: By comparing the marketing to sales ratios, we can identify the soft drink products that generate higher sales with relatively lower marketing investment. These products might be more popular or have better market penetration and can be prioritized for further growth. Analyzing underperforming products: If a specific product has a high marketing to sales ratio compared to others in the portfolio, it might indicate an underperforming product. We can investigate the reasons behind this underperformance, such as inadequate marketing strategies, poor product positioning, or high competition.
- Compare caffeinated and non-caffeinated beverage sales: Analyze the data to determine which category has higher sales, and calculate the marketing-to-sales ratio for each category.

3. Business Question(s):

1. What product gave most amount of sales?, Marketing to sales ratio of the product?
2. Are caffinated drinks gives more sales than non caffinated drinks?, Marketing to sales ratio of the caffinated vs non caffinated beverages?
3. Which product has highest Gross Profit Margin?

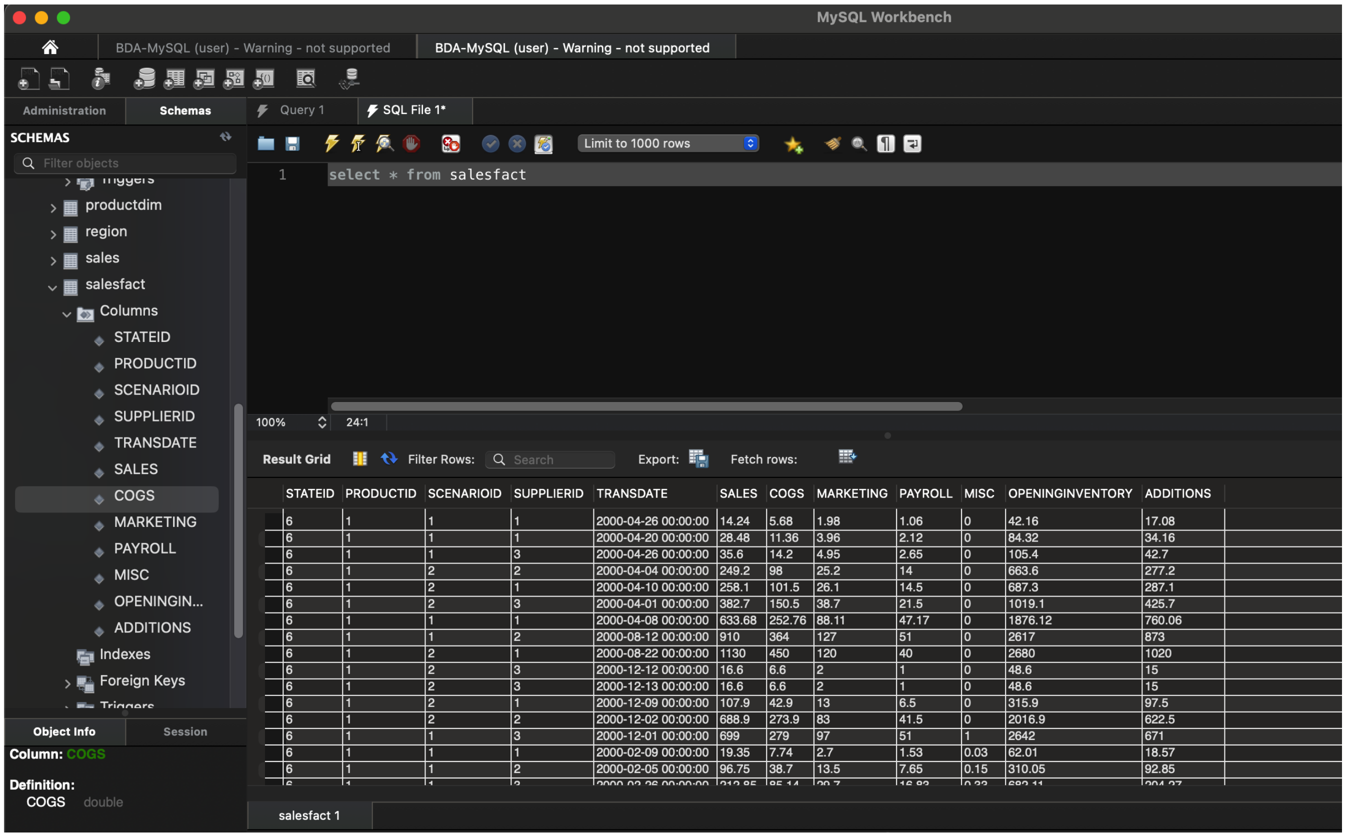
4. Data Exploration:

Data exploration is the process of discovering patterns, trends, and insights in a dataset. SQL is a powerful tool for data exploration because it allows users to retrieve and analyze large amounts of data quickly and efficiently.

To begin data exploration in SQL, the first step is to identify the dataset to be analyzed. This may involve querying a database or importing data from an external source.

In our case we are extracting data from instance made on google cloud.

Next steps is to run basic queries to have a deeper look at our data and to run queries to find if we have NULL values

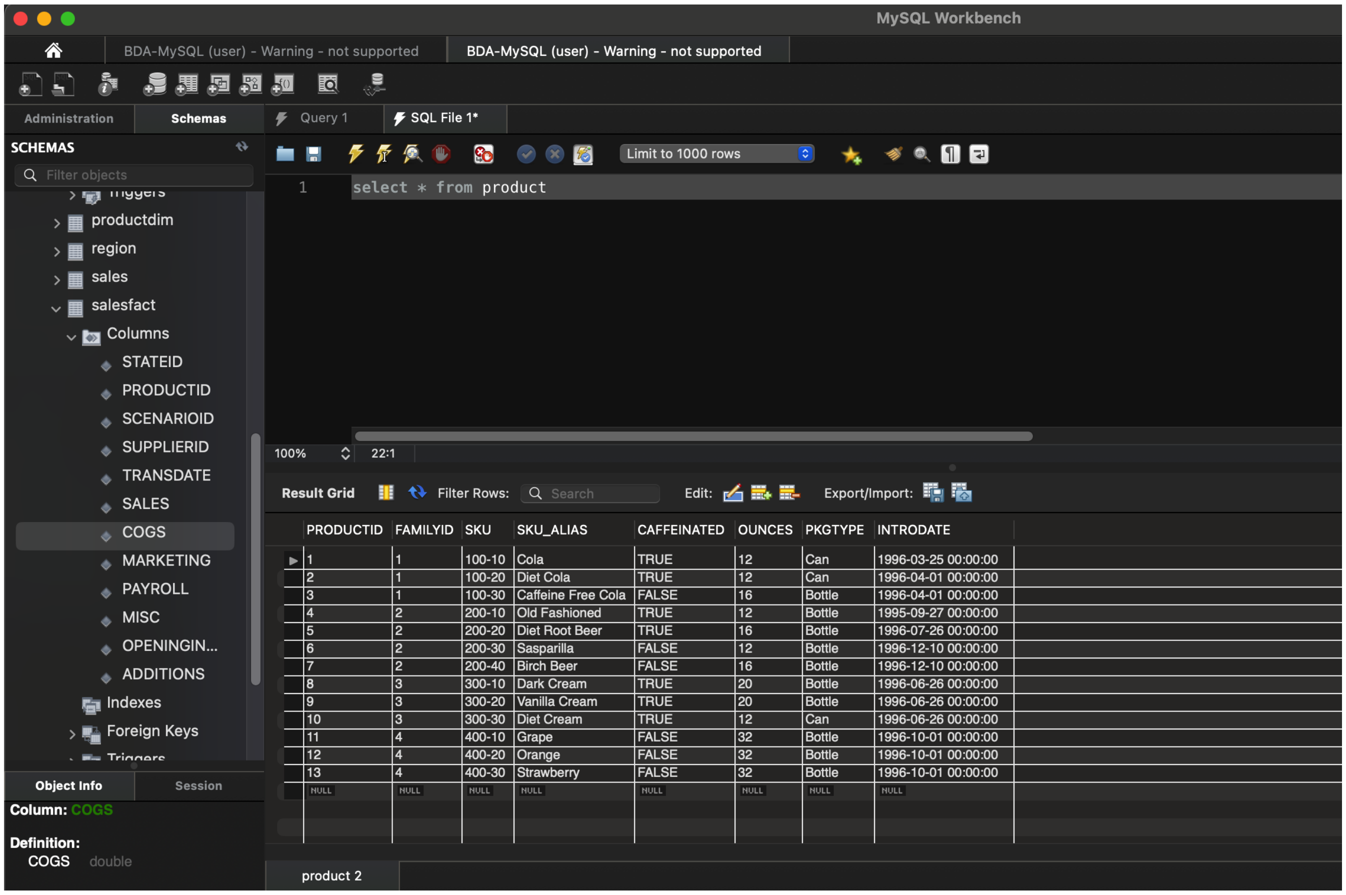


```
select * from salesfact
```

Column: COGS

Definition:
COGS double

STATEID	PRODUCTID	SCENARIOID	SUPPLIERID	TRANSDATE	SALES	COGS	MARKETING	PAYROLL	MISC	OPENINGINVENTORY	ADDITIONS
6	1	1	1	2000-04-26 00:00:00	14.24	5.68	1.98	1.06	0	42.16	17.08
6	1	1	1	2000-04-20 00:00:00	28.48	11.36	3.96	2.12	0	84.32	34.16
6	1	1	3	2000-04-26 00:00:00	35.6	14.2	4.95	2.65	0	105.4	42.7
6	1	2	2	2000-04-04 00:00:00	249.2	98	25.2	14	0	663.6	277.2
6	1	2	1	2000-04-10 00:00:00	258.1	101.5	26.1	14.5	0	687.3	287.1
6	1	2	3	2000-04-01 00:00:00	382.7	150.5	38.7	21.5	0	1019.1	425.7
6	1	1	1	2000-04-08 00:00:00	633.68	252.76	88.11	47.17	0	1876.12	760.06
6	1	1	2	2000-08-12 00:00:00	910	364	127	51	0	2617	873
6	1	2	1	2000-08-22 00:00:00	1130	450	120	40	0	2680	1020
6	1	2	3	2000-12-12 00:00:00	16.6	6.6	2	1	0	48.6	15
6	1	2	3	2000-12-13 00:00:00	16.6	6.6	2	1	0	48.6	15
6	1	2	1	2000-12-09 00:00:00	107.9	42.9	13	6.5	0	315.9	97.5
6	1	2	2	2000-12-02 00:00:00	688.9	273.9	83	41.5	0	2016.9	622.5
6	1	1	3	2000-12-01 00:00:00	699	279	97	51	1	2642	671
6	1	1	1	2000-02-09 00:00:00	19.35	7.74	2.7	1.53	0.03	62.01	18.57
6	1	1	2	2000-02-05 00:00:00	96.75	38.7	13.5	7.65	0.15	310.05	92.85



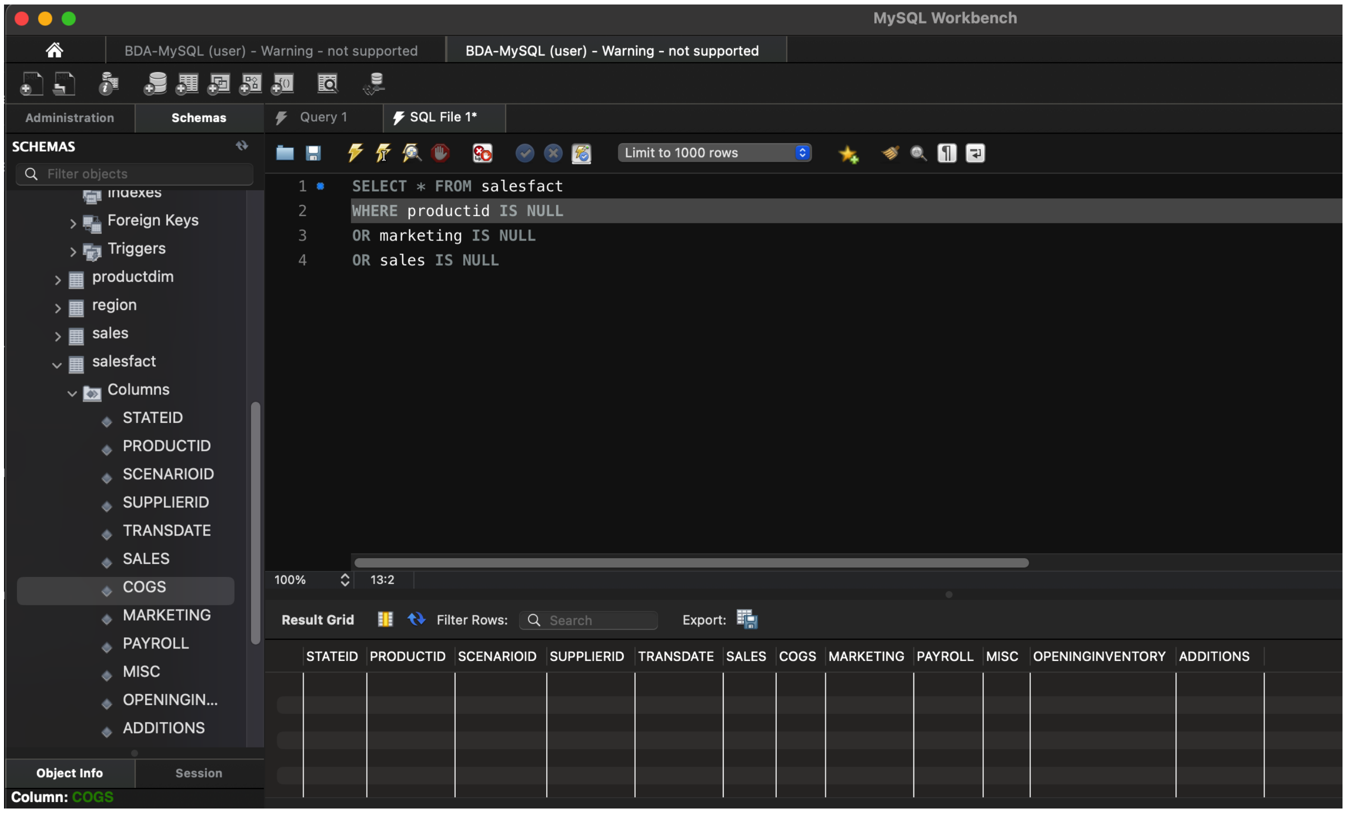
```
select * from product
```

Column: COGS

Definition:
COGS double

PRODUCTID	FAMILYID	SKU	SKU_ALIAS	CAFFEINATED	OUNCES	PKGTYPE	INTRODATE
1	1	100-10	Cola	TRUE	12	Can	1996-03-25 00:00:00
2	1	100-20	Diet Cola	TRUE	12	Can	1996-04-01 00:00:00
3	1	100-30	Caffeine Free Cola	FALSE	16	Bottle	1996-04-01 00:00:00
4	2	200-10	Old Fashioned	TRUE	12	Bottle	1995-09-27 00:00:00
5	2	200-20	Diet Root Beer	TRUE	16	Bottle	1996-07-26 00:00:00
6	2	200-30	Sasparilla	FALSE	12	Bottle	1996-12-10 00:00:00
7	2	200-40	Birch Beer	FALSE	16	Bottle	1996-12-10 00:00:00
8	3	300-10	Dark Cream	TRUE	20	Bottle	1996-06-26 00:00:00
9	3	300-20	Vanilla Cream	TRUE	20	Bottle	1996-06-26 00:00:00
10	3	300-30	Diet Cream	TRUE	12	Can	1996-06-26 00:00:00
11	4	400-10	Grape	FALSE	32	Bottle	1996-10-01 00:00:00
12	4	400-20	Orange	FALSE	32	Bottle	1996-10-01 00:00:00
13	4	400-30	Strawberry	FALSE	32	Bottle	1996-10-01 00:00:00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Loading [MathJax]/extensions/Safe.js



```
SELECT * FROM salesfact
WHERE productid IS NULL
OR marketing IS NULL
OR sales IS NULL
```

STATEID	PRODUCTID	SCENARIOID	SUPPLIERID	TRANSDATE	SALES	COGS	MARKETING	PAYROLL	MISC	OPENINGINVENTORY	ADDITIONS

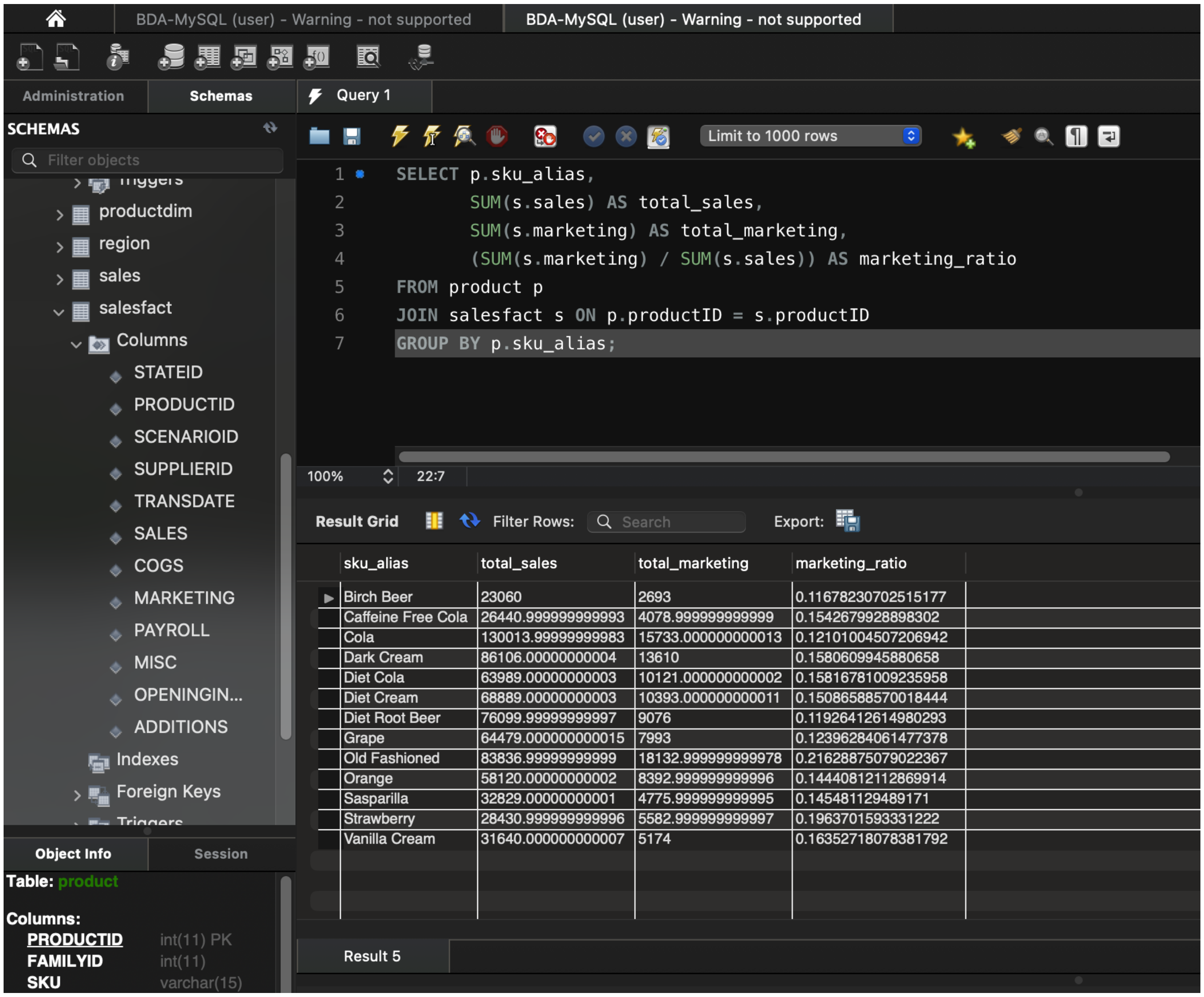
This code is performing a SELECT query on a table called "salesfact".

The WHERE clause is used to filter the results of the SELECT query based on certain conditions. In this case, it is filtering the results to include only the rows where at least one of the following conditions is true:

"productid" column is NULL "marketing" column is NULL "sales" column is NULL This means that the query will return all the rows from the "salesfact" table where either the "productid", "marketing", or "sales" column has a NULL value.

5. Database Queries & Results:

Loading [MathJax]/extensions/Safe.js



```sql
SELECT p.sku_alias,
       SUM(s.sales) AS total_sales,
       SUM(s.marketing) AS total_marketing,
       (SUM(s.marketing) / SUM(s.sales)) AS marketing_ratio
FROM product p
JOIN salesfact s ON p.productID = s.productID
GROUP BY p.sku_alias;
```

Table: product

Columns:

PRODUCTID	int(11) PK
FAMILYID	int(11)
SKU	varchar(15)

sku_alias	total_sales	total_marketing	marketing_ratio
Birch Beer	23060	2693	0.11678230702515177
Caffeine Free Cola	26440.999999999993	4078.999999999999	0.1542679928898302
Cola	130013.99999999983	15733.000000000013	0.12101004507206942
Dark Cream	86106.00000000004	13610	0.1580609945880658
Diet Cola	63989.00000000003	10121.000000000002	0.15816781009235958
Diet Cream	68889.00000000003	10393.000000000011	0.15086588570018444
Diet Root Beer	76099.99999999997	9076	0.11926412614980293
Grape	64479.000000000015	7993	0.12396284061477378
Old Fashioned	83836.99999999999	18132.999999999978	0.21628875079022367
Orange	58120.00000000002	8392.999999999996	0.14440812112869914
Sasparilla	32829.00000000001	4775.999999999995	0.145481129489171
Strawberry	28430.999999999996	5582.999999999997	0.1963701593331222
Vanilla Cream	31640.000000000007	5174	0.16352718078381792

This SQL query retrieves and calculates the sales and marketing data for each product SKU alias. Here's a breakdown of what the code does:

1. SELECT: The query selects four columns to be displayed in the final result: sku_alias, total_sales, total_marketing, and marketing_ratio.
2. p.sku_alias: This column represents the SKU alias of a product from the product table.
3. SUM(s.sales) AS total_sales: This part of the query calculates the total sales for each SKU alias by summing the sales column from the salesfact table and assigning an alias total_sales to the resulting column.
4. SUM(s.marketing) AS total_marketing: This part calculates the total marketing expenditure for each SKU alias by summing the marketing column from the salesfact table and assigning an alias total_marketing to the resulting column.
5. (SUM(s.marketing) / SUM(s.sales)) AS marketing_ratio: This part calculates the marketing ratio by dividing the total marketing expenditure by the total sales for each SKU alias. It assigns an alias marketing_ratio to the resulting column.
6. FROM product p: This clause specifies that the query should use the product table and assigns the alias p to the table.

Loading [MathJax]/extensions/Safe.js

7. JOIN salesfact s ON p.productID = s.productID: This line performs an inner join between the product table (aliased as p) and the salesfact table (aliased as s) based on the condition that their productID columns match.

8. GROUP BY p.sku_alias: This clause groups the results by the sku_alias column from the product table, aggregating the sales and marketing data for each SKU alias.

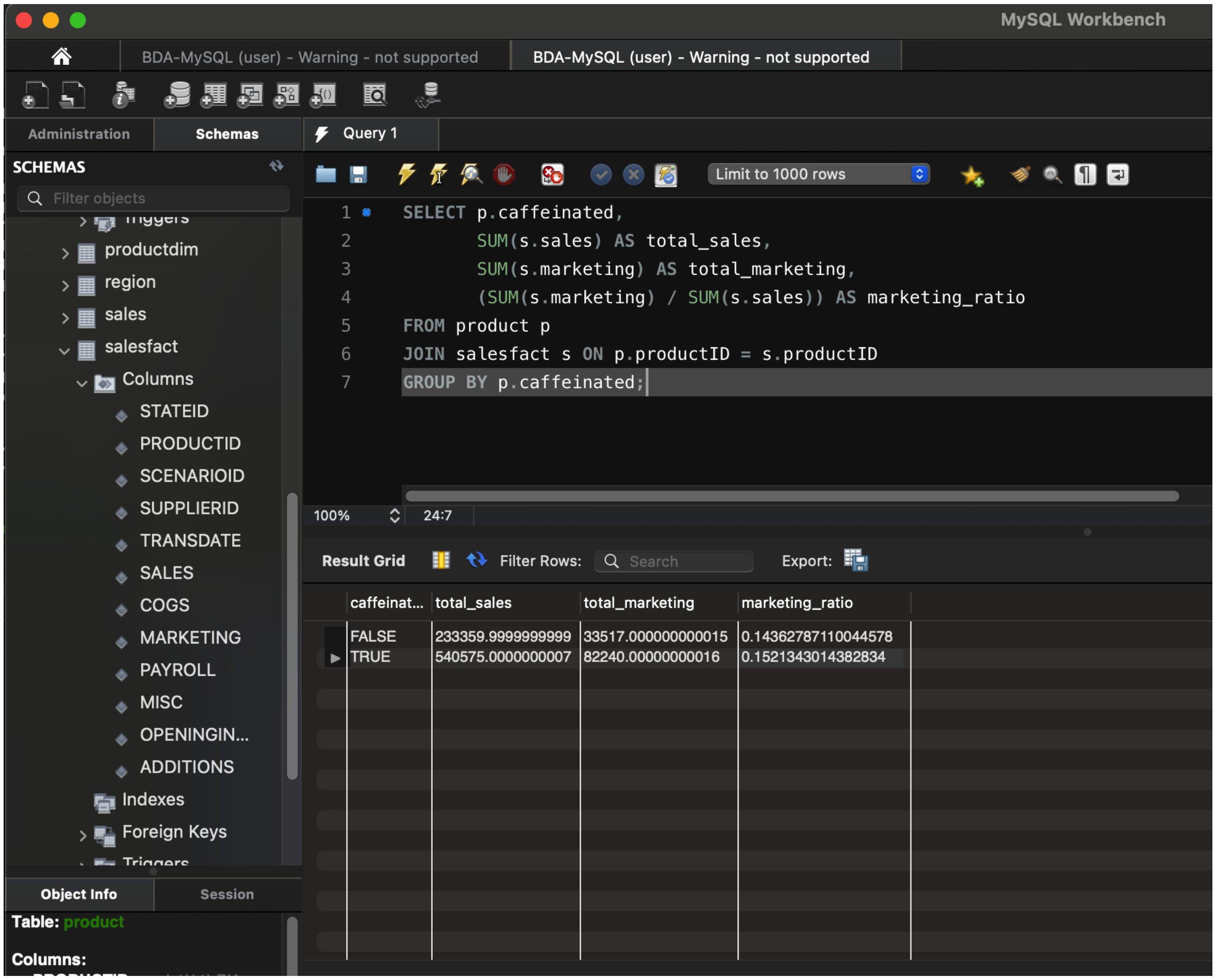
This table shows the sales and marketing data for various soft drink products. Here's a brief analysis of the results:

Total Sales: Cola has the highest total sales at 130,014 units, while Strawberry has the lowest total sales at 28,431 units.

Total Marketing: Old Fashioned has the highest total marketing expenditure at 18,133, while Birch Beer has the lowest at 2,693.

Marketing Ratio: The marketing ratio is calculated as the total marketing expenditure divided by the total sales. It represents the proportion of sales generated from marketing activities. The higher the marketing ratio, the more effective the marketing efforts are assumed to be.

Old Fashioned has the highest marketing ratio at 0.216289, meaning it has the most effective marketing efforts among the products. Birch Beer has the lowest marketing ratio at 0.116782, indicating that its marketing efforts are the least effective among the products. In summary, Cola is the most popular product in terms of total sales, while Old Fashioned has the most effective marketing efforts as indicated by the highest marketing ratio. On the other hand, Strawberry has the lowest sales, and Birch Beer has the least effective marketing efforts based on the marketing ratio.



```sql
SELECT p.caffeinated,
       SUM(s.sales) AS total_sales,
       SUM(s.marketing) AS total_marketing,
       (SUM(s.marketing) / SUM(s.sales)) AS marketing_ratio
FROM product p
JOIN salesfact s ON p.productID = s.productID
GROUP BY p.caffeinated;
```

caffeinat...	total_sales	total_marketing	marketing_ratio
FALSE	233359.9999999999	33517.000000000015	0.14362787110044578
TRUE	540575.0000000007	82240.00000000016	0.1521343014382834

The SQL query is selecting data from two tables product and salesfact and joining them on productID column. The selected columns include the caffeinated column from product table, the SUM of sales and marketing columns from salesfact table, and a calculated column marketing_ratio which is the ratio of total_marketing to total_sales.

The result of the query is grouped by the caffeinated column from product table, which means that the resulting DataFrame will have one row per unique value of the caffeinated column, and the aggregated values for the other selected columns for each group.

Based on the provided data, it appears that caffeinated beverages have higher total sales and higher marketing expenditure compared to non-caffeinated beverages. Here is a summary of the data:

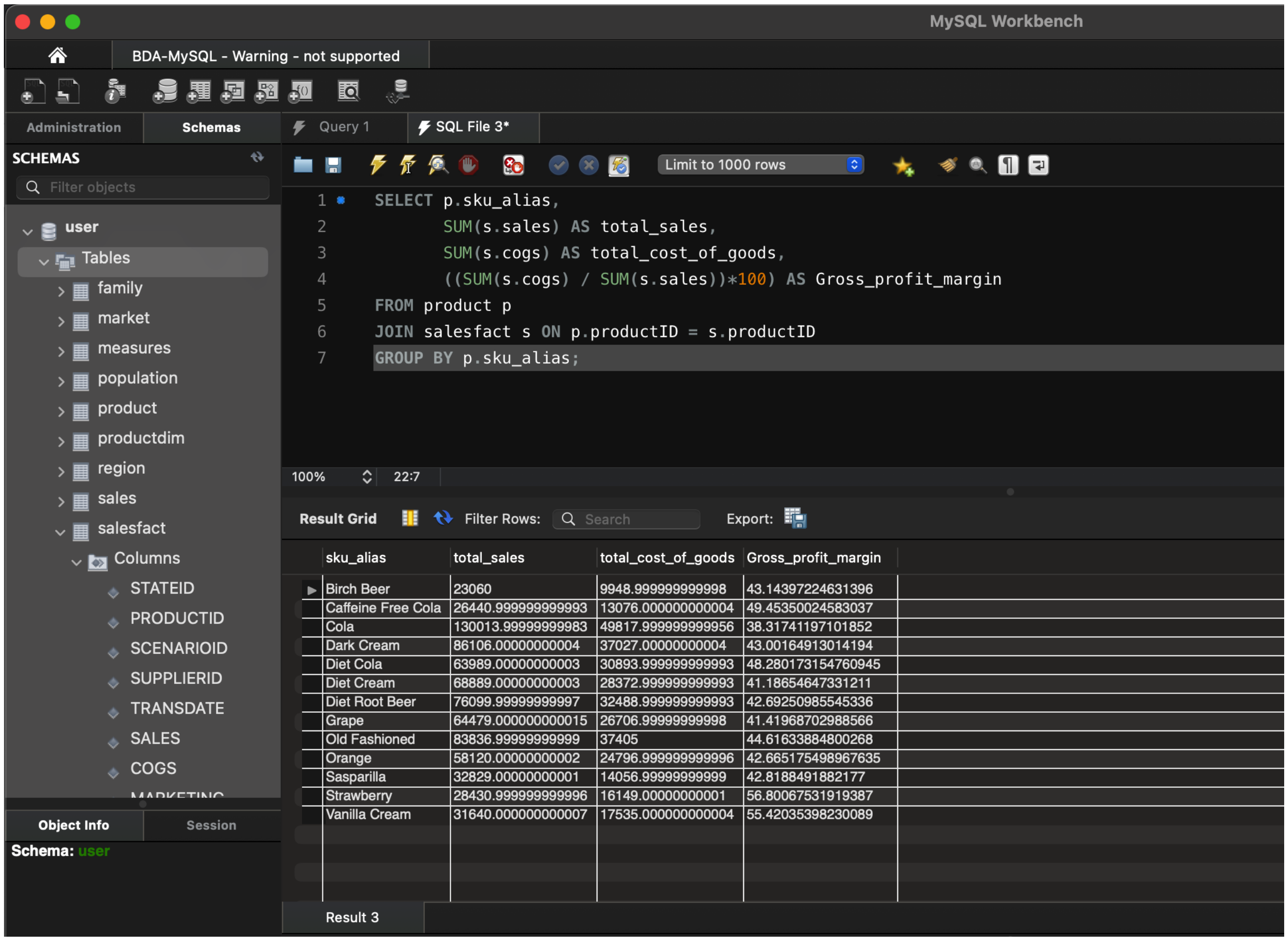
- Non-caffeinated beverages (0):
- Total sales: 233,360
- Total marketing: 33,517
- Marketing ratio: 14.36%

- Caffeinated beverages (1):

- Total sales: 540,575
- Total marketing: 82,240
- Marketing ratio: 15.21%

Loading [MathJax]/extensions/Safe.js

A potential business decision could be to focus more on caffeinated beverages, as they are generating higher sales. However, it's important to consider the higher marketing ratio for caffeinated beverages (15.21% vs 14.36% for non-caffeinated). This means that a larger percentage of sales revenue is spent on marketing for caffeinated beverages.



```
SELECT p.sku_alias,
       SUM(s.sales) AS total_sales,
       SUM(s.cogs) AS total_cost_of_goods,
       ((SUM(s.cogs) / SUM(s.sales))*100) AS Gross_profit_margin
FROM product p
JOIN salesfact s ON p.productID = s.productID
GROUP BY p.sku_alias;
```

sku_alias	total_sales	total_cost_of_goods	Gross_profit_margin
Birch Beer	23060	9948.999999999998	43.14397224631396
Caffeine Free Cola	26440.999999999993	13076.000000000004	49.45350024583037
Cola	130013.99999999983	49817.999999999956	38.31741197101852
Dark Cream	86106.00000000004	37027.00000000004	43.00164913014194
Diet Cola	63989.00000000003	30893.999999999993	48.280173154760945
Diet Cream	68889.00000000003	28372.999999999993	41.18654647331211
Diet Root Beer	76099.99999999997	32488.999999999993	42.69250985545336
Grape	64479.000000000015	26706.99999999998	41.41968702988566
Old Fashioned	83836.99999999999	37405	44.61633884800268
Orange	58120.00000000002	24796.999999999996	42.665175498967635
Sasparilla	32829.00000000001	14056.99999999999	42.8188491882177
Strawberry	28430.999999999996	16149.00000000001	56.80067531919387
Vanilla Cream	31640.000000000007	17535.000000000004	55.42035398230089

This query joins two tables, "product" and "salesfact", to calculate the total sales, total cost of goods sold, and gross profit margin for each product SKU.

The query selects the product's SKU alias and uses the JOIN clause to link it with the salesfact table on the productID column.

The GROUP BY clause groups the data by SKU alias, which means that the query will return a row for each unique SKU alias in the product table.

The SUM function is used to calculate the total sales and total cost of goods sold for each SKU. The formula for gross profit margin is then applied, which is the cost of goods sold divided by sales, multiplied by 100 to get a percentage.

Overall, this query provides valuable insights into a company's sales performance by calculating the profitability of each product SKU.

This data provides an overview of the sales performance and profitability of various soda flavors. It includes information on total sales, total cost of goods, and gross profit margin for each flavor. Here's a brief summary of the insights we can draw from this data:

Loading [MathJax]/extensions/Safe.js

- Cola is the best-selling flavor with total sales of 130,014 units, followed by Dark Cream (86,106 units) and Diet Root Beer (76,100 units).
- Sasparilla has the lowest total sales with only 32,829 units sold, followed by Vanilla Cream (31,640 units) and Strawberry (28,431 units).
- Strawberry has the highest gross profit margin at 56.80%, followed by Vanilla Cream (55.42%) and Caffeine Free Cola (49.45%).
- Cola has the lowest gross profit margin at 38.32%, which might indicate that it is priced more competitively or has higher production costs compared to the other flavors.

In conclusion, this data helps us understand which soda flavors are most popular, as well as their profitability. Cola seems to be driving sales, but it has a lower gross profit margin compared to other flavors. On the other hand, Strawberry and Vanilla Cream have the highest gross profit margins but lower sales. This information can be useful for the company to make decisions on marketing, pricing, and production strategies to maximize profitability and cater to consumer preferences.

6. Analysis:

In [95]:
```
import mysql.connector
import pandas as pd
from pandas import *
import sqlite3
import pprint as pp
from pyspark.sql.functions import *
import matplotlib.pyplot as plt
```

In [96]:
```
cnx = mysql.connector.connect(user='user', password= 'T4MPZmVun', host = '35.211.35.48',
```

In [72]:
```
query = """
SELECT p.sku_alias,
       SUM(s.sales) AS total_sales,
       SUM(s.marketing) AS total_marketing,
       (SUM(s.marketing) / SUM(s.sales)) AS marketing_ratio
FROM product p
JOIN salesfact s ON p.productID = s.productID
GROUP BY p.sku_alias;
"""

df = pd.read_sql(query, cnx)
df
```

Loading [MathJax]/extensions/Safe.js

Out[72]:

	sku_alias	total_sales	total_marketing	marketing_ratio
0	Birch Beer	23060.0	2693.0	0.116782
1	Caffeine Free Cola	26441.0	4079.0	0.154268
2	Cola	130014.0	15733.0	0.121010
3	Dark Cream	86106.0	13610.0	0.158061
4	Diet Cola	63989.0	10121.0	0.158168
5	Diet Cream	68889.0	10393.0	0.150866
6	Diet Root Beer	76100.0	9076.0	0.119264
7	Grape	64479.0	7993.0	0.123963
8	Old Fashioned	83837.0	18133.0	0.216289
9	Orange	58120.0	8393.0	0.144408
10	Sasparilla	32829.0	4776.0	0.145481
11	Strawberry	28431.0	5583.0	0.196370
12	Vanilla Cream	31640.0	5174.0	0.163527

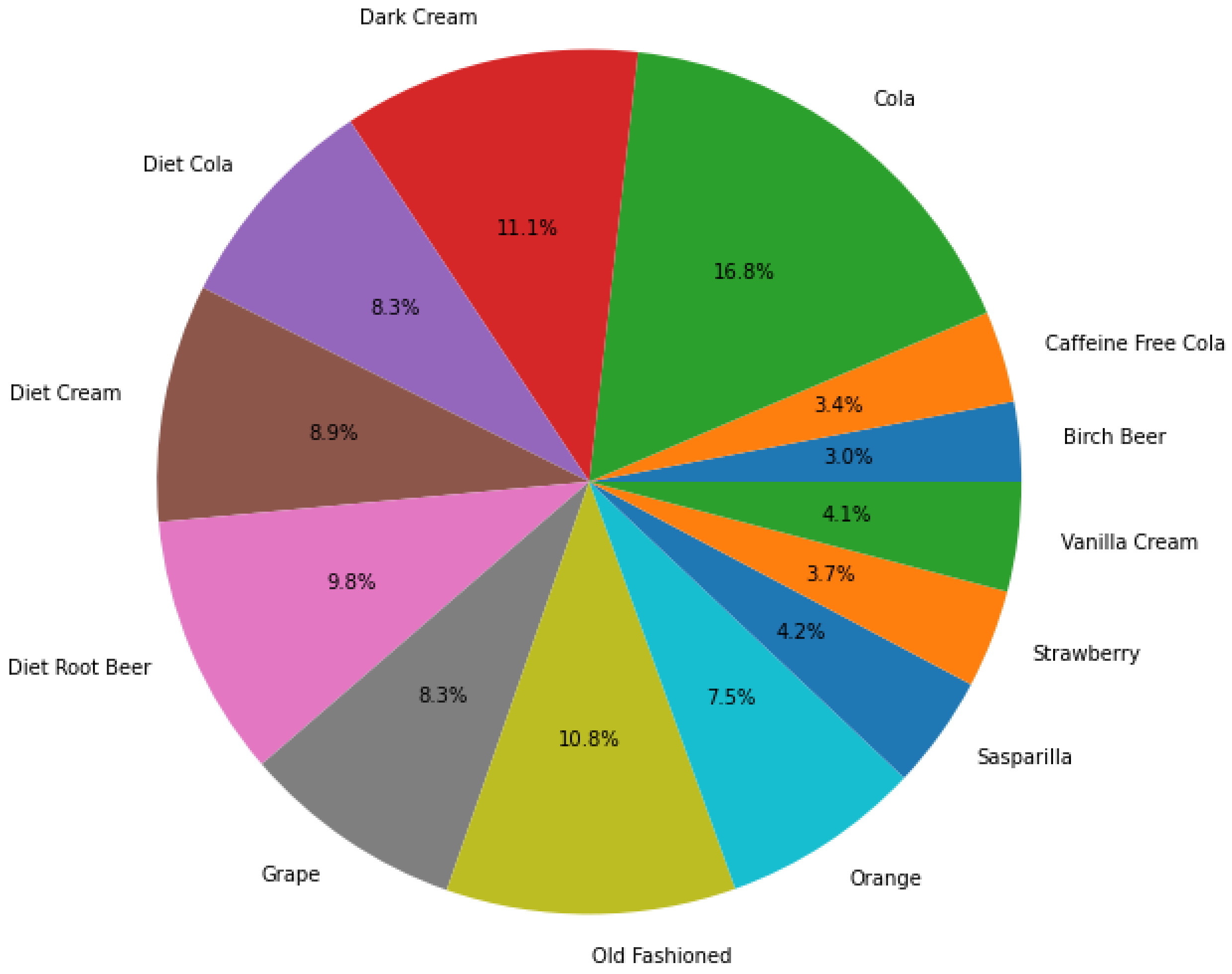
In [78]:

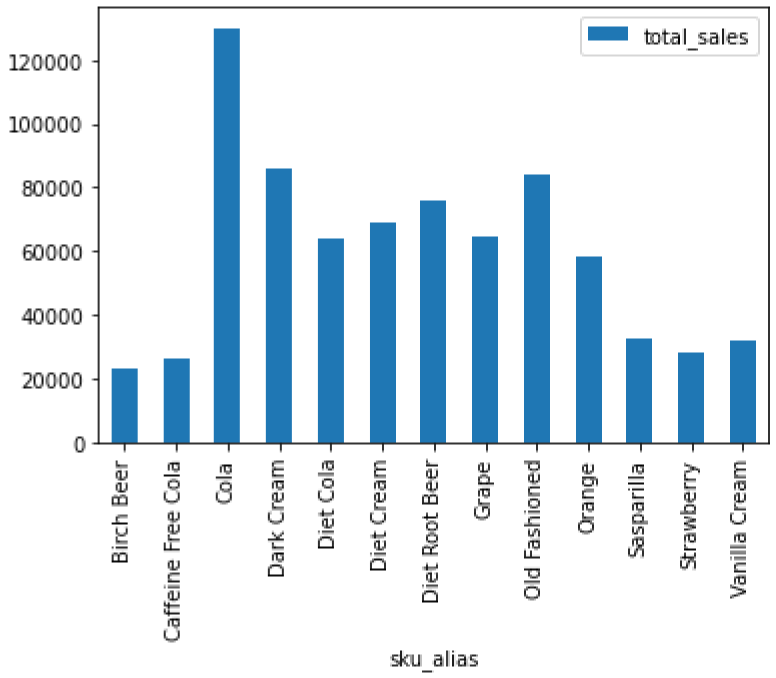
```
data = df['total_sales']
labels = df['sku_alias']
# Create a 1x2 grid of subplots
#plt.subplot(1, 2, 1)
plt.figure(figsize = (10,10))
plt.pie(data, labels=labels, autopct='%1.1f%%')

#plt.subplot(1, 2, 2)
df.plot(kind='bar', x='sku_alias', y='total_sales')
```

Out[78]:

```
<AxesSubplot:xlabel='sku_alias'>
```

Loading [MathJax]/extensions/Safe.js





In [97]:
```
query = """
SELECT p.caffeinated,
       SUM(s.sales) AS total_sales,
       SUM(s.marketing) AS total_marketing,
       (SUM(s.marketing) / SUM(s.sales)) AS marketing_ratio
FROM product p
JOIN salesfact s ON p.productID = s.productID
```
Loading [MathJax]/extensions/Safe.js

```
GROUP BY p.caffeinated;
"""

df1 = pd.read_sql(query, cnx)
df1
```

Out[97]:

	caffeinated	total_sales	total_marketing	marketing_ratio
0	FALSE	233360.0	33517.0	0.143628
1	TRUE	540575.0	82240.0	0.152134

In [105…

```
data = {'caffeinated': [False, True], 'total_amount': [233360.0, 540575.0], 'marketing_r
]}
df1 = pd.DataFrame(data)
df1['caffeinated'] = df1['caffeinated'].apply(lambda x: 'Caffeinated' if x else 'Not Caf
print(df1)
```

```
       caffeinated  total_amount  marketing_ratio
0  Not Caffeinated      233360.0         0.143628
1      Caffeinated      540575.0         0.152134
```

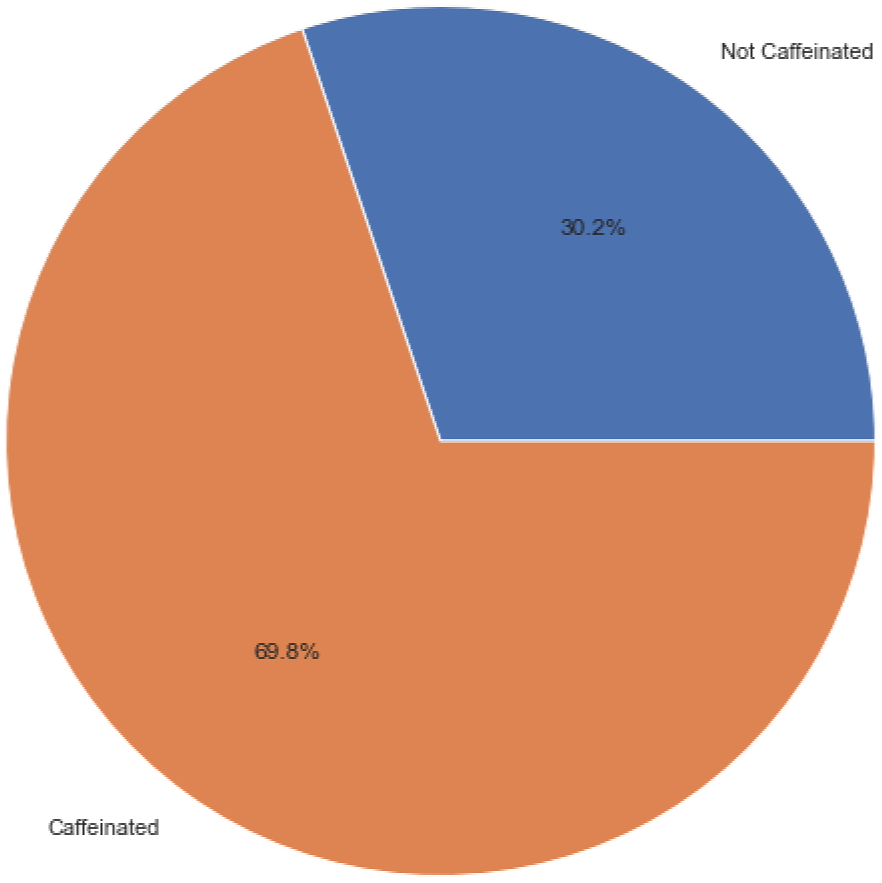
In [106…

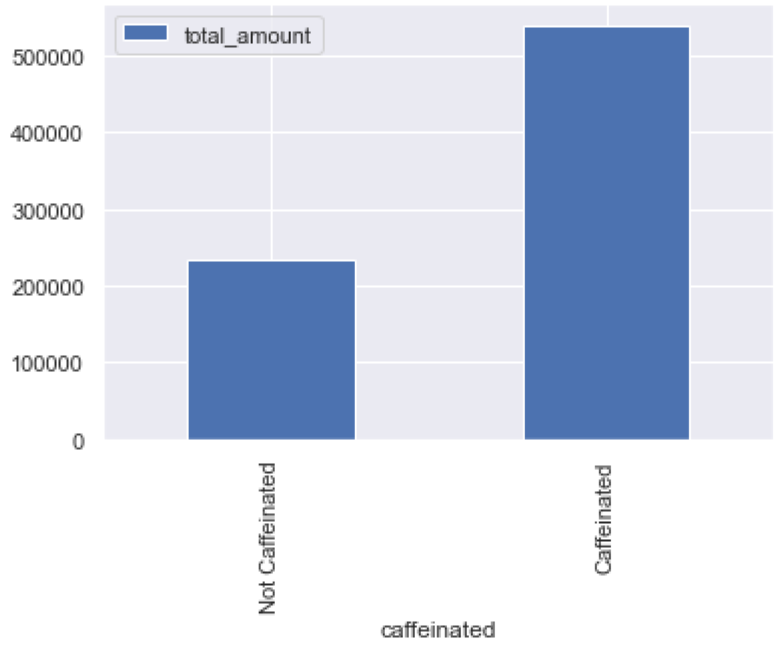
```
df1['caffeinated']
data = df1['total_amount']
labels = df1['caffeinated']
# Create a 1x2 grid of subplots
#plt.subplot(1, 2, 1)
plt.figure(figsize = (10,10))
plt.pie(data, labels=labels, autopct='%1.1f%%')

#plt.subplot(1, 2, 2)
df1.plot(kind='bar', x='caffeinated', y='total_amount')
```

Out[106]:

```
<AxesSubplot:xlabel='caffeinated'>
```

Loading [MathJax]/extensions/Safe.js





In [84]:
```
query = """
SELECT p.sku_alias,
       SUM(s.sales) AS total_sales,
       SUM(s.cogs) AS total_cost_of_goods,
       ((SUM(s.cogs) / SUM(s.sales))*100) AS Gross_profit_margin
FROM product p
JOIN salesfact s ON p.productID = s.productID
GROUP BY p.sku_alias;
```

Loading [MathJax]/extensions/Safe.js

```
"""

df2 = pd.read_sql(query, cnx)
df2
```

```
/Users/farazahmed/opt/anaconda3/lib/python3.9/site-packages/pandas/io/sql.py:761: UserWa
rning: pandas only support SQLAlchemy connectable(engine/connection) ordatabase string U
RI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please consider usin
g SQLAlchemy
  warnings.warn(
```

Out[84]:

	sku_alias	total_sales	total_cost_of_goods	Gross_profit_margin
0	Birch Beer	23060.0	9949.0	43.143972
1	Caffeine Free Cola	26441.0	13076.0	49.453500
2	Cola	130014.0	49818.0	38.317412
3	Dark Cream	86106.0	37027.0	43.001649
4	Diet Cola	63989.0	30894.0	48.280173
5	Diet Cream	68889.0	28373.0	41.186546
6	Diet Root Beer	76100.0	32489.0	42.692510
7	Grape	64479.0	26707.0	41.419687
8	Old Fashioned	83837.0	37405.0	44.616339
9	Orange	58120.0	24797.0	42.665175
10	Sasparilla	32829.0	14057.0	42.818849
11	Strawberry	28431.0	16149.0	56.800675
12	Vanilla Cream	31640.0	17535.0	55.420354

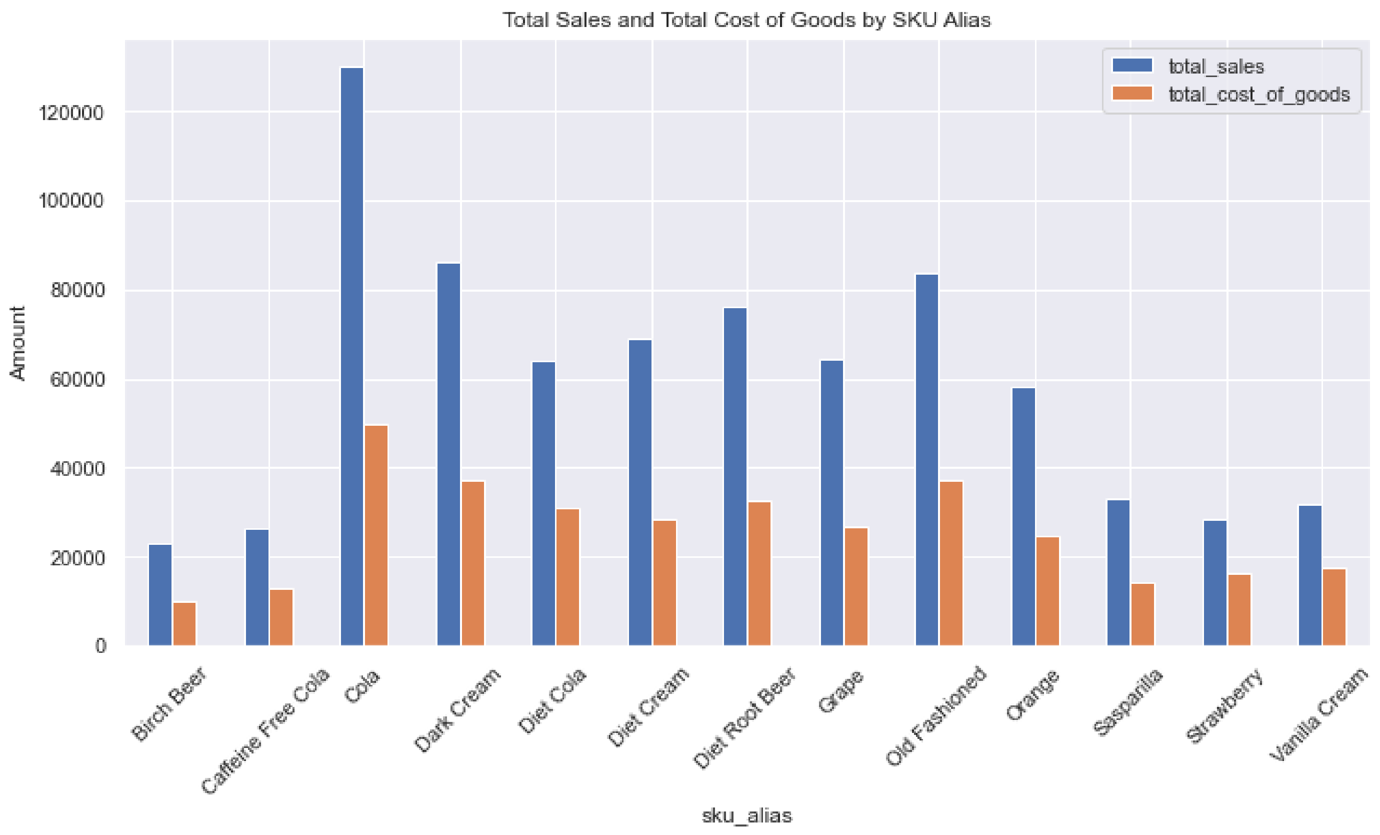
In [90]:

```
# Plot bar chart
ax = df2.plot.bar(x='sku_alias', y=['total_sales', 'total_cost_of_goods'], figsize=(12,
ax.set_ylabel('Amount')
ax.set_title('Total Sales and Total Cost of Goods by SKU Alias')
```

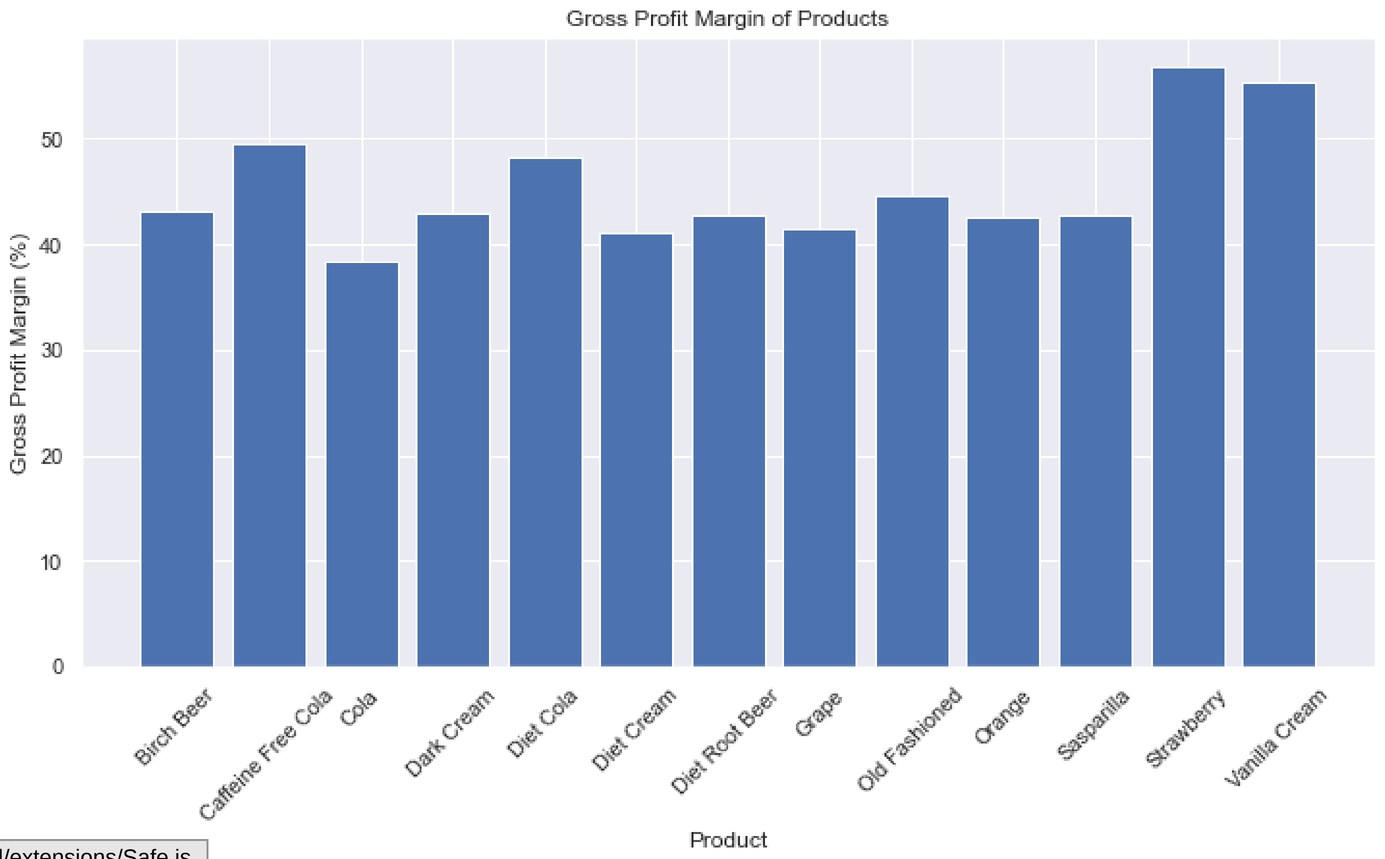
Out[90]:

```
Text(0.5, 1.0, 'Total Sales and Total Cost of Goods by SKU Alias')
```

Loading [MathJax]/extensions/Safe.js



```
In [101…  # Bar chart
         plt.figure(figsize=(12, 6))
         plt.bar(df2['sku_alias'], df2['Gross_profit_margin'])
         plt.xticks(rotation=45)
         plt.xlabel('Product')
         plt.ylabel('Gross Profit Margin (%)')
         plt.title('Gross Profit Margin of Products')

         # Show the plot
         plt.show()
```



Loading [MathJax]/extensions/Safe.js

7. Decisions:

Based on our analysis, here are some business decisions that can help reduce cost and increase efficiency:

1. Focus on high gross-profit-margin products: Products such as Strawberry and Vanilla Cream have the highest gross-profit-margins (56.80% and 55.42% respectively). Focusing on these products can help increase overall profit margins while reducing costs associated with lower-margin products.

2. Re-evaluate marketing strategies for products with high marketing ratios: Products like Old Fashioned and Strawberry have higher marketing ratios (0.216289 and 0.196370 respectively). Investigate the effectiveness of marketing strategies for these products and consider reallocating marketing budgets to products with lower marketing ratios and higher profit margins.

3. Optimize marketing strategies based on caffeination: The second data set shows that Caffeinated products have a slightly higher marketing ratio (0.152134) than Not Caffeinated products (0.143628). Analyze the marketing strategies for both categories to determine if a different approach can be used to improve efficiency and reduce costs.

4. Explore cost reduction opportunities in the supply chain: Analyze the cost of goods for each product to identify potential savings. Negotiate better deals with suppliers, or explore alternative suppliers to lower the total cost of goods.

5. Review product assortment and consider discontinuing low-performing products: Evaluate product performance based on total sales and profit margins. Consider discontinuing or reducing the focus on products with low sales and low profit margins, such as Birch Beer and Caffeine Free Cola. This can help free up resources to focus on more profitable products.

In summary, focusing on high gross-profit-margin products, optimizing marketing strategies, exploring cost reduction opportunities in the supply chain, and reviewing the product assortment can help your business reduce costs and increase efficiency.

8. Conclusions:

In conclusion, based on the data and analysis, our business can reduce costs and increase efficiency by implementing the following strategies:

- Concentrate on high gross-profit-margin products such as Strawberry and Vanilla Cream to maximize profits.
- Re-evaluate and optimize marketing strategies for products with high marketing ratios, reallocating marketing budgets to products with lower marketing ratios and higher profit margins.
- Analyze and adjust marketing strategies based on caffeination levels to improve efficiency and reduce costs.
- Investigate cost reduction opportunities in the supply chain by negotiating better deals with suppliers or exploring alternative suppliers to lower the total cost of goods.
- Review the product assortment, considering the discontinuation or reduction of focus on low-performing products to free up resources for more profitable products.

By adopting these strategies, our business can achieve a more cost-effective and efficient operation, ultimately resulting in improved profitability and growth.