



SQL

blinkit Analysis

SQL PROJECT
ON BLINKIT SALES ANALYSIS

blinkit Analysis

SQL

INTRODUCTION

- ★ Blinkit is a quick-commerce company delivering groceries and essentials in minutes.
- ★ This project uses SQL to analyze sales data across products, outlets, and customers.
- ★ It focuses on KPIs like **Total Sales, Average Sales, Items Sold, and Customer Ratings**.
- ★ Raw sales data is transformed into **actionable insights for business decisions**.
- ★ The study highlights **trends, patterns, and key drivers of Blinkit's performance**.

blinkit Analysis

SQL

OBJECTIVE

- ★ To identify **top-performing products and item categories**.
- ★ To analyze **sales distribution across outlets and locations**.
- ★ To measure the effect of **fat content, outlet size, and establishment year** on sales.
- ★ To study **regional trends** across Tier 1, Tier 2, and Tier 3 markets.
- ★ To deliver insights that improve **sales, marketing, and customer satisfaction**.



SQL

blinkit Analysis

BUSINESS REQUIREMENT

- 1. Total Sales:** The overall revenue generated from all items sold.
- 2. Average Sales:** The average revenue per sale.
- 3. Number of Items:** The total count of different items sold.
- 4. Average Rating:** The average customer rating for items sold.

blinkit Analysis

BUSINESS REQUIREMENT

1. Total Sales by Fat Content:

Objective: Analyze the impact of fat content on total sales.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.

2. Total Sales by Item Type:

Objective: Identify the performance of different item types in terms of total sales.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.

3. Fat Content by Outlet for Total Sales:

Objective: Compare total sales across different outlets segmented by fat content.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.

4. Total Sales by Outlet Establishment:

Objective: Evaluate how the age or type of outlet establishment influences total sales.



SQL

blinkit Analysis

BUSINESS REQUIREMENT

5. Percentage of Sales by Outlet Size:

Objective: Analyze the correlation between outlet size and total sales.

6. Sales by Outlet Location:

Objective: Assess the geographic distribution of sales across different locations.

7. All Metrics by Outlet Type:

Objective: Provide a comprehensive view of all key metrics (Total Sales, Average Sales, Number of Items, Average Rating) broken down by different outlet types.



SQL

blinkit Analysis

CREATE TABLE

```
DROP TABLE IF EXISTS blinkit_sales;  
  
CREATE TABLE blinkit_sales (  
    id SERIAL , -- auto-generated unique id  
    item_identifier VARCHAR,  
    item_fat_content VARCHAR,  
    item_type VARCHAR,  
    outlet_establishment_year INT,  
    outlet_identifier VARCHAR,  
    outlet_location_type VARCHAR,  
    outlet_size VARCHAR,  
    outlet_type VARCHAR,  
    item_visibility FLOAT,  
    item_weight FLOAT,  
    total_sales FLOAT,  
    rating FLOAT  
);
```

 SQL

blinkit Analysis

TABLE

Query Query History

```
41 );
42
43 select * from blinkit_sales;
44
45
46
```

Data Output Messages Notifications

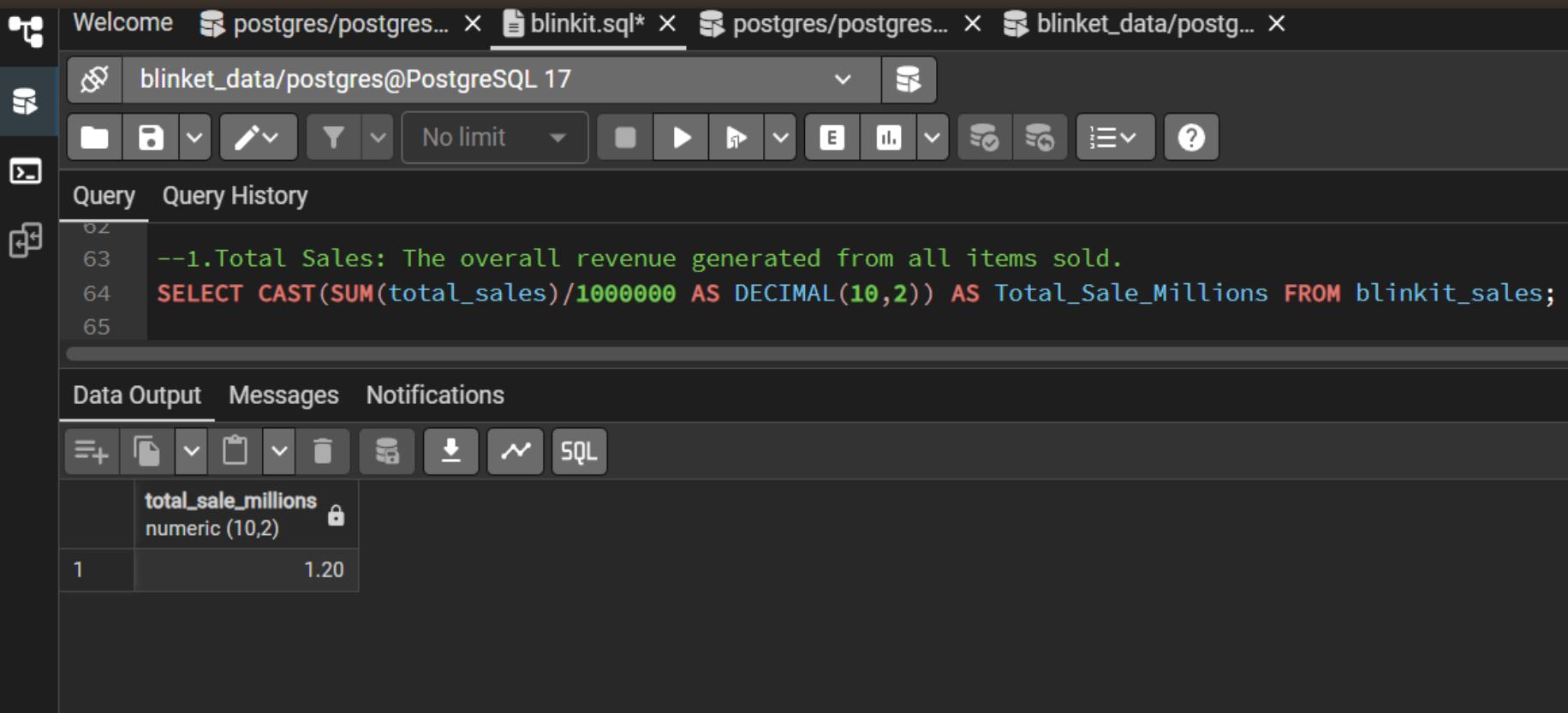
Showing rows: 1 to 1000 | [Edit](#) | Page No: 1 of 9 | [First](#) [Previous](#) [Next](#) [Last](#)

	id	integer	item_identifier	character varying	item_fat_content	character varying	item_type	character varying	outlet_establishment_year	integer	outlet_identifier	character varying	outlet_location_type	character varying	outlet_size	character varying	outlet_type	character varying	item_visibility	double precision	item_weight	double precision	total_doub
1		1284	FDE51	Regular	Dairy				2011	OUT010	Tier 3		Small	Grocery Store		0.161466534			5.925				
2		18	NCB07	Low Fat	Household				2012	OUT049	Tier 1		Medium	Supermarket Type1		0.077628053			19.2				
3		78	FDU49	Regular	Canned				2012	OUT049	Tier 1		Medium	Supermarket Type1		0.030742083			19.5				
4		4437	FDK27	Low Fat	Meat				1998	OUT019	Tier 1		Small	Grocery Store		0.015664229			[null]				
5		4517	FDB28	Low Fat	Dairy				2011	OUT010	Tier 3		Medium	Grocery Store		0.156307983			6.615				
6		4627	FD003	Regular	Meat				1998	OUT019	Tier 1		Small	Grocery Store		0.064577332			[null]				
7		79	FDA02	Regular	Dairy				2012	OUT049	Tier 1		Medium	Supermarket Type1		0.02976887			14				
8		80	FDV26	Regular	Dairy				2012	OUT049	Tier 1		Medium	Supermarket Type1		0			20.25				
9		140	FD001	Regular	Breakfast				2010	OUT046	Tier 1		Small	Supermarket Type1		0.020718655			21.1				
10		1	FDX32	Regular	Fruits and Vegetables				2012	OUT049	Tier 1		Medium	Supermarket Type1		0.1000135			15.1				
11		2	NCB42	Low Fat	Health and Hygiene				2022	OUT018	Tier 3		Medium	Supermarket Type2		0.008596051			11.8				
12		3	FDR28	Regular	Frozen Foods				2010	OUT046	Tier 1		Small	Supermarket Type1		0.025896485			13.85				
13		4	FDL50	Regular	Canned				2000	OUT013	Tier 3		High	Supermarket Type1		0.04277867			12.15				
14		5	DRI25	Low Fat	Soft Drinks				2015	OUT045	Tier 2		Small	Supermarket Type1		0.033970195			19.6				

 SQL

blinkit Analysis

Total Sales: The overall revenue generated from all items sold.



Welcome postgres/postgres... × blinkit.sql* × postgres/postgres... × blinket_data/postg... ×

blinket_data/postgres@PostgreSQL 17

No limit

Query History

```
--1.Total Sales: The overall revenue generated from all items sold.  
SELECT CAST(SUM(total_sales)/1000000 AS DECIMAL(10,2)) AS Total_Sale_Millions FROM blinkit_sales;
```

Data Output Messages Notifications

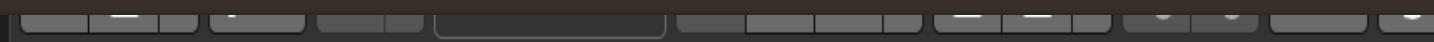
total_sale_millions

	total_sale_millions
1	1.20

 SQL

blinkit Analysis

Average Sales: The average revenue per sale.



Query Query History

```
65  
66 --2.Average Sales: The average revenue per sale.  
67 SELECT CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale FROM blinkit_sales;  
68  
69
```

Data Output Messages Notifications



	average_sale
	numeric (10) 

1	141
---	-----

SQL

blinkit Analysis

Number of Items: The total count of different items sold.

The screenshot shows a SQL development environment with the following interface elements:

- Toolbar:** Includes icons for file operations (New, Open, Save, etc.), search, filters, and various execution and monitoring tools.
- Query History:** Shows the history of queries run, with the current query being numbered 71.
- Current Query:** Contains the SQL code:

```
--3.Number of Items: The total count of different items sold.  
SELECT COUNT(*) AS No_of_item FROM blinkit_sales;
```
- Data Output:** A table showing the results of the query:

	no_of_item
1	8523
- Bottom Bar:** Includes icons for file operations, a refresh button, a download button, and a SQL tab.

 SQL

blinkit Analysis

Average Rating: The average customer rating for items sold.



Query Query History

```
74  
75 --4.Average Rating: The average customer rating for items sold.  
76 SELECT CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating FROM blinkit_sales;  
77  
78
```

Data Output Messages Notifications



	avg_rating
	numeric (10,1)
1	4.0

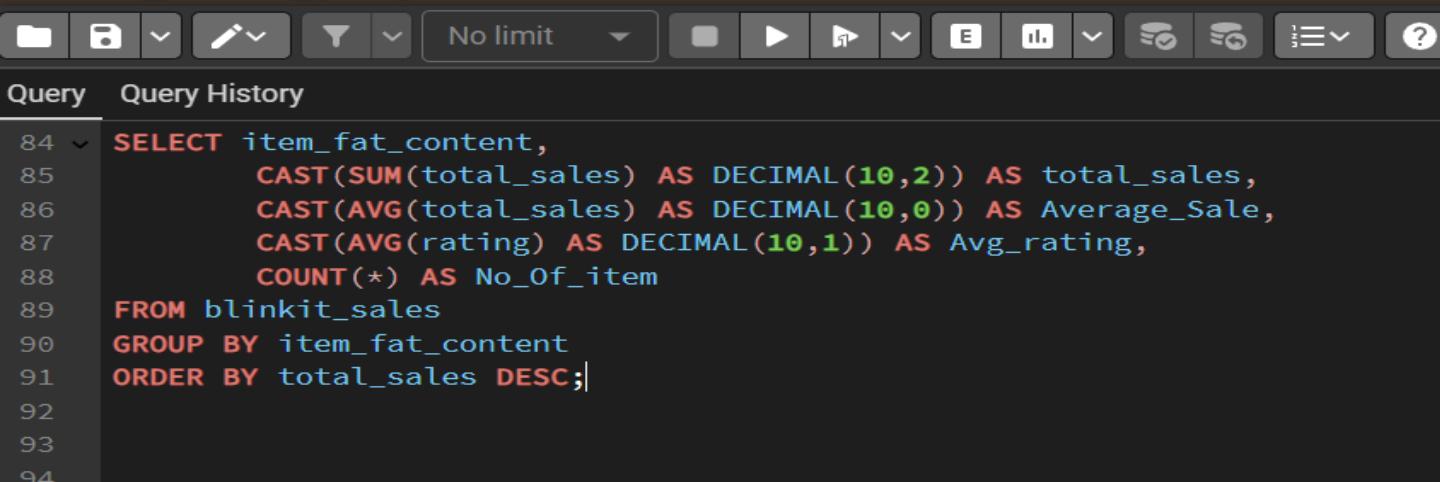
 SQL

blinkit Analysis

Total Sales by Fat Content:

Objective: Analyze the impact of fat content on total sales.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.



```
84 SELECT item_fat_content,
85     CAST(SUM(total_sales) AS DECIMAL(10,2)) AS total_sales,
86     CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
87     CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
88     COUNT(*) AS No_Of_item
89 FROM blinkit_sales
90 GROUP BY item_fat_content
91 ORDER BY total_sales DESC;
```

Data Output Messages Notifications



SQL

	item_fat_content	total_sales	average_sale	avg_rating	no_of_item
1	Low Fat	776319.68	141	4.0	5517
2	Regular	425361.80	142	4.0	3006



blinkit Analysis

Total Sales by Item Type:

Objective: Identify the performance of different item types in terms of total sales.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.

```
98
99  SELECT item_type ,
100      CAST(SUM(total_sales) AS DECIMAL(10,2)) AS Total_Sales_By_Item ,
101      CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
102      CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
103      COUNT(*) AS No_Of_item
104  FROM blinkit_Sales
105  GROUP BY item_type
106  ORDER BY Total_Sales_By_Item DESC
107  LIMIT 5;
```

Data Output						
	item_type	total_sales_by_item	average_sale	avg_rating	no_of_item	
1	Fruits and Vegetables	178124.08	145	4.0	1232	
2	Snack Foods	175433.92	146	3.9	1200	
3	Household	135976.53	149	4.0	910	
4	Frozen Foods	118558.88	139	4.0	856	
5	Dairy	101276.46	148	4.0	682	

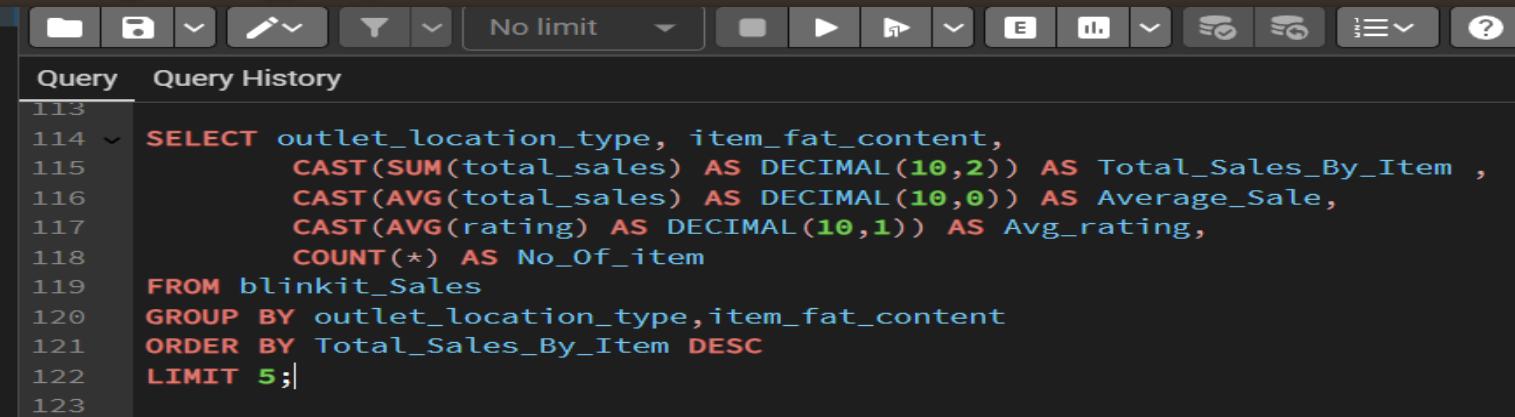
 SQL

blinkit Analysis

Fat Content by Outlet for Total Sales:

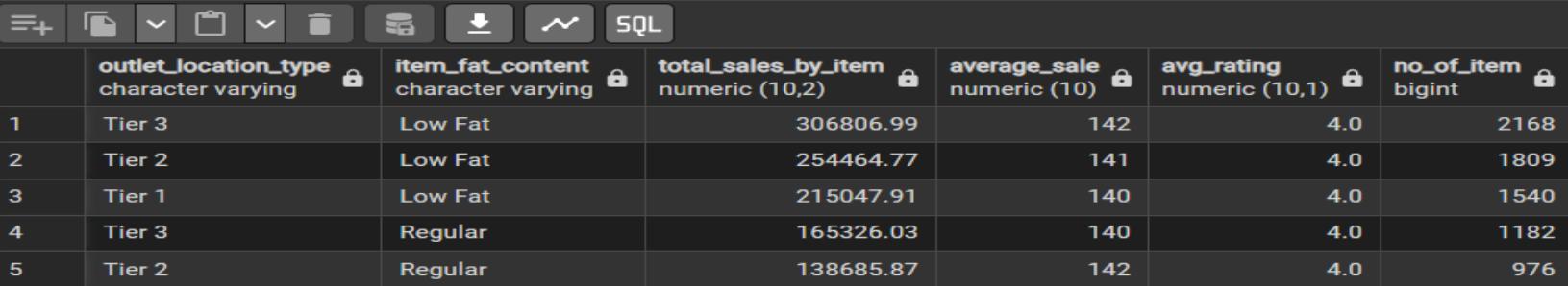
Objective: Compare total sales across different outlets segmented by fat content.

Additional KPI Metrics: Assess how other KPIs (Average Sales, Number of Items, Average Rating) vary with fat content.



```
113
114 SELECT outlet_location_type, item_fat_content,
115     CAST(SUM(total_sales) AS DECIMAL(10,2)) AS Total_Sales_By_Item ,
116     CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
117     CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
118     COUNT(*) AS No_of_item
119 FROM blinkit_Sales
120 GROUP BY outlet_location_type,item_fat_content
121 ORDER BY Total_Sales_By_Item DESC
122 LIMIT 5;|
```

Data Output Messages Notifications



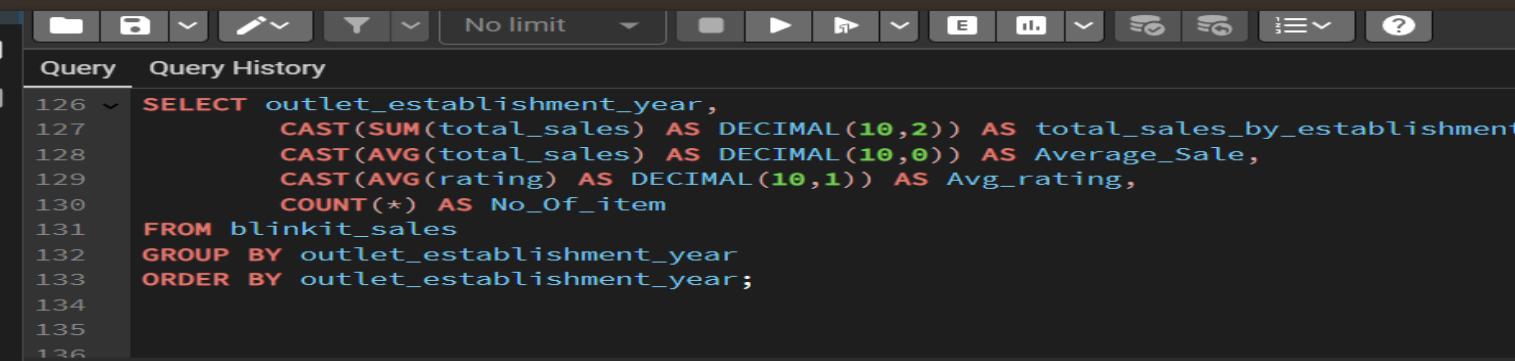
	outlet_location_type character varying	item_fat_content character varying	total_sales_by_item numeric (10,2)	average_sale numeric (10)	avg_rating numeric (10,1)	no_of_item bigint
1	Tier 3	Low Fat	306806.99	142	4.0	2168
2	Tier 2	Low Fat	254464.77	141	4.0	1809
3	Tier 1	Low Fat	215047.91	140	4.0	1540
4	Tier 3	Regular	165326.03	140	4.0	1182
5	Tier 2	Regular	138685.87	142	4.0	976

 SQL

blinkit Analysis

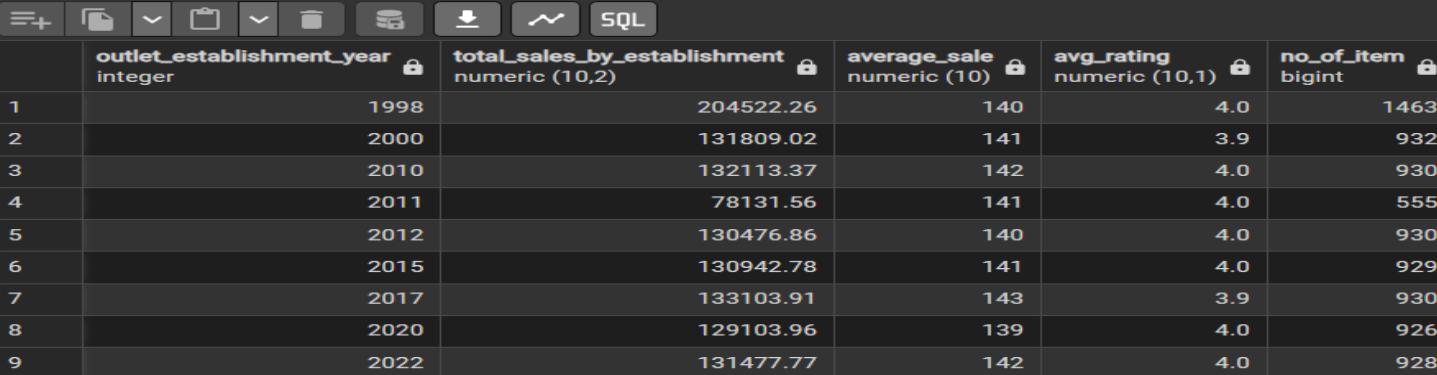
Total Sales by Outlet Establishment:

Objective: Evaluate how the age or type of outlet establishment influences total sales



```
126    SELECT outlet_establishment_year,
127          CAST(SUM(total_sales) AS DECIMAL(10,2)) AS total_sales_by_establishment,
128          CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
129          CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
130          COUNT(*) AS No_Of_item
131     FROM blinkit_sales
132    GROUP BY outlet_establishment_year
133   ORDER BY outlet_establishment_year;
```

Data Output Messages Notifications



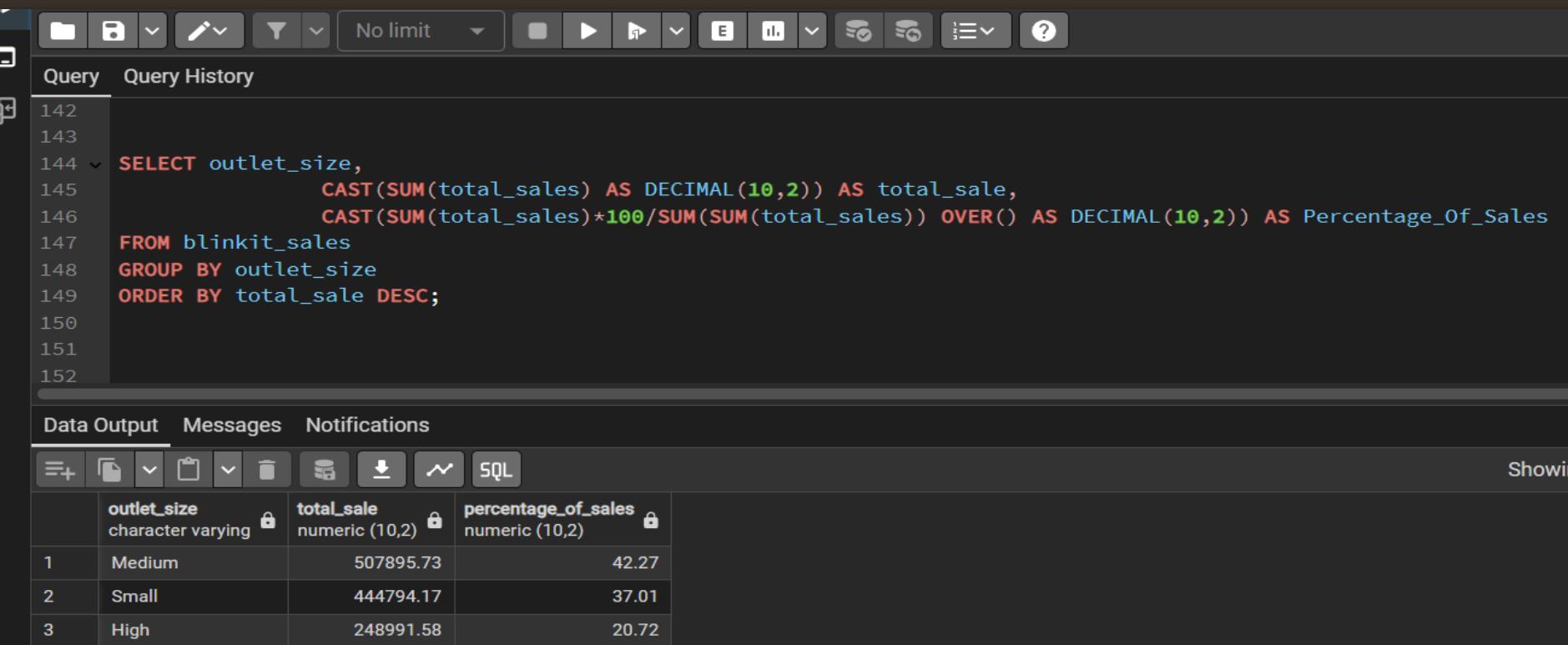
	outlet_establishment_year	total_sales_by_establishment	average_sale	avg_rating	no_of_item
1	1998	204522.26	140	4.0	1463
2	2000	131809.02	141	3.9	932
3	2010	132113.37	142	4.0	930
4	2011	78131.56	141	4.0	555
5	2012	130476.86	140	4.0	930
6	2015	130942.78	141	4.0	929
7	2017	133103.91	143	3.9	930
8	2020	129103.96	139	4.0	926
9	2022	131477.77	142	4.0	928

 SQL

blinkit Analysis

Percentage of Sales by Outlet Size:

Objective: Analyze the correlation between outlet size and total sales.



The screenshot shows a SQL development environment with the following interface elements:

- Toolbar:** Includes icons for file operations (New, Open, Save, Print), search, and various database management functions.
- Query Editor:** Shows a code editor with a numbered code block. The code uses windowed aggregation to calculate the percentage of sales for each outlet size relative to the total sales across all sizes.
- Output Tab:** Labeled "Data Output", showing the results of the query.
- Message Tab:** Labeled "Messages".
- Notification Tab:** Labeled "Notifications".
- Bottom Bar:** Includes icons for file operations (New, Open, Save, Print, Export, Import, Refresh, Help), a "SQL" tab, and a status message "Showing n".

```
142
143
144 SELECT outlet_size,
145         CAST(SUM(total_sales) AS DECIMAL(10,2)) AS total_sale,
146         CAST(SUM(total_sales)*100/SUM(SUM(total_sales)) OVER() AS DECIMAL(10,2)) AS Percentage_of_Sales
147     FROM blinkit_sales
148     GROUP BY outlet_size
149     ORDER BY total_sale DESC;
150
151
152
```

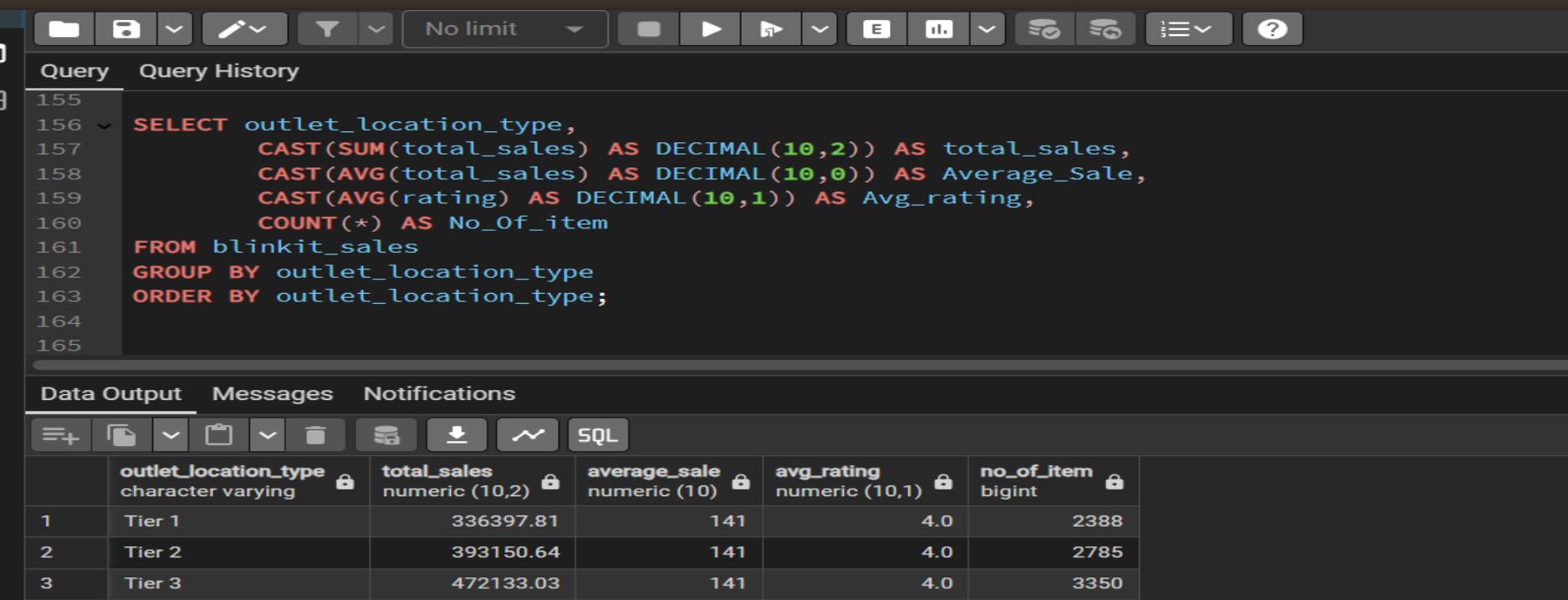
	outlet_size character varying	total_sale numeric (10,2)	percentage_of_sales numeric (10,2)
1	Medium	507895.73	42.27
2	Small	444794.17	37.01
3	High	248991.58	20.72

 SQL

blinkit Analysis

Sales by Outlet Location:

Objective: Assess the geographic distribution of sales across different locations.



The screenshot shows a SQL interface with the following details:

Query History:

```
155
156 SELECT outlet_location_type,
157     CAST(SUM(total_sales) AS DECIMAL(10,2)) AS total_sales,
158     CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
159     CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
160     COUNT(*) AS No_Of_item
161 FROM blinkit_sales
162 GROUP BY outlet_location_type
163 ORDER BY outlet_location_type;
164
165
```

Data Output:

	outlet_location_type	total_sales	average_sale	avg_rating	no_of_item
1	Tier 1	336397.81	141	4.0	2388
2	Tier 2	393150.64	141	4.0	2785
3	Tier 3	472133.03	141	4.0	3350



blinkit Analysis

All Metrics by Outlet Type:

Objective: Provide a comprehensive view of all key metrics (Total Sales, Average Sales, Number of Items, Average Rating) broken down by different outlet types.

```
169
170 ✓ SELECT outlet_type,
171     CAST(SUM(total_sales) AS DECIMAL(10,2)) AS total_sales,
172     CAST(AVG(total_sales) AS DECIMAL(10,0)) AS Average_Sale,
173     CAST(AVG(rating) AS DECIMAL(10,1)) AS Avg_rating,
174     COUNT(*) AS No_of_item
175 FROM blinkit_sales
176 GROUP BY outlet_type
177 ORDER BY total_sales DESC;
```

Data Output		Messages		Notifications	
	outlet_type character varying		total_sales numeric (10,2)		average_sale numeric (10)
1	Supermarket Type1		787549.89		141
2	Grocery Store		151939.15		140
3	Supermarket Type2		131477.77		142
4	Supermarket Type3		130714.67		140

blinkit Analysis

SQL

ADVANTAGES

- ★ Helps Blinkit identify **best-selling and low-selling items**.
- ★ Supports **inventory planning and supply chain optimization**.
- ★ Guides **business expansion** by highlighting profitable outlets and regions.
- ★ Improves **customer experience** by linking ratings with sales data.
- ★ Builds a **data-driven foundation** for dashboards and future analytics.



SQL

blinkit Analysis

THANK YOU