

Blinkit Analysis



Description

In this project, I performed comprehensive data analysis on the Blinkit Grocery dataset using Microsoft SQL Server. The dataset contains detailed information about item fat contain, item identifier, item type, total sales etc. My objective was to extract meaningful insights that could help a business like Blinkit improve its product strategy, outlet performance, and customer satisfaction.



I created and executed multiple SQL queries to answer key business questions such as:

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.

5. Total Sales by Fat content:

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

6. Total sales by item Type:

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

7. Fat Content by Outlet for Total Sales:

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

8. Total Sales by outlet Establishment:

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

9. percentage of Sales by Outlet Size:

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

10. Sales by Outlet Location:

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

11. 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.

- 12. Top 5 items with the highest Sales.**
- 13. Outlet which has the highest total Sales.**
- 14. Top 3 best_ selling item types in Tier 3 locatins.**
- 15. Outlet type which sells the most number of distinct items.**
- 16. Showing yearly trends : average sales per establishment year of outlets.**
- 17. The average Sales by item fat content and item type.**

- 18. the outlet that sells the most variety of items.**
- 19. the top 3 outlet types by total sales.**
- 20. The heaviest items by average weight per item type.**
- 21. Items that are rarely visible but having high sales.**



1. TOTAL SALES:

```
select cast(sum(Total_Sales)/1000000.0 as decimal(10,2)) as  
Total_Sales_Million  
from blinkit_data;
```

Results Messages

	Total_Sales_Million
1	1.20

2. AVERAGE SALES

```
select cast(avg(Total_Sales) as int) as Avg_Sales  
from blinkit_data;
```

Results Messages

	Avg_Sales
1	140

3. No. of Items:

```
select count(*) as No_of_Orders  
from blinkit_data;
```

Results Messages

	No_of_Orders
1	8523

4. AVG RATING:

```
select cast(avg(Rating) as decimal(10,1)) as Avg_Rating  
from blinkit_data;
```

Results Messages

	Avg_Rating
1	4.0

5. Total Sales by Fat Content:

```
select Item_Fat_Content, cast(sum(Total_sales) as decimal(10,2)) as  
Total_Sales  
From blinkit_data  
group by Item_Fat_Content
```

	Item_Fat_Content	Total_Sales
1	Low Fat	776319.68
2	Regular	425361.80

6. Total Sales by Item Type

```
select Item_Type, cast(sum(Total_Sales) as decimal(10,2)) as Total_Sales  
from blinkit_data  
group by Item_Type  
order by Total_Sales desc
```



	Item_Type	Total_Sales
1	Fruits and Vegetables	178124.08
2	Snack Foods	175433.92
3	Household	135976.53
4	Frozen Foods	118558.88
5	Dairy	101276.46
6	Canned	90706.73
7	Baking Goods	81894.74
8	Health and Hygiene	68025.84
9	Meat	59449.86
10	Soft Drinks	58514.16
11	Breads	35379.12
12	Hard Drinks	29334.68
13	Others	22451.89
14	Starchy Foods	21880.03
15	Breakfast	15596.70
16	Seafood	9077.87

7. Fat Content by Outlet for Total Sales:

```
select Outlet_Location_Type,
       isnull([Low Fat], 0) as Low_Fat,
       isnull([Regular], 0) as Regular
  from (
    select Outlet_Location_Type, Item_Fat_Content,
           cast(sum(Total_Sales) as decimal(10,2)) as Total_Sales
      from blinkit_data
     group by Outlet_Location_Type, Item_Fat_Content
  ) as SourceTable
 pivot
(
  sum(Total_Sales)
  for Item_Fat_Content in ([Low Fat],[Regular])
) as PivotTable
order by Outlet_Location_Type;
```

	Outlet_Location_Type	Low_Fat	Regular
1	Tier 1	215047.91	121349.90
2	Tier 2	254464.77	138685.87
3	Tier 3	306806.99	165326.03

8. Total Sales by Outlet Establishment:

```
select Outlet_Establishment_Year, cast(sum(Total_Sales) as decimal(10,2))  
as Total_Sales  
from blinkit_data  
group by Outlet_Establishment_Year  
order by Outlet_Establishment_Year
```

	Outlet_Establishment_Year	Total_Sales
1	1998	204522.26
2	2000	131809.02
3	2010	132113.37
4	2011	78131.56
5	2012	130476.86
6	2015	130942.78
7	2017	133103.91
8	2020	129103.96
9	2022	131477.77

9. Percentage of Sales by Outlet

Size:

```
select
    Outlet_Size,
    cast(sum(Total_Sales) as decimal(10,2)) as Total_sales,
    cast((sum(Total_Sales)*100.0/sum(sum(Total_Sales))) over() as
decimal(10,2)) as Sales_Percentage
from blinkit_data
group by Outlet_Size
order by Total_Sales desc;
```

The screenshot shows a SQL query results window with two tabs: 'Results' and 'Messages'. The 'Results' tab displays a table with four columns: 'Outlet_Size', 'Total_sales', and 'Sales_Percentage'. The data is as follows:

	Outlet_Size	Total_sales	Sales_Percentage
1	Medium	507895.73	42.27
2	Small	444794.17	37.01
3	High	248991.58	20.72

10. Sales by Outlet Location:

```
select Outlet_Location_Type, cast(sum(Total_Sales) as decimal(10,2)) as  
Total_Sales  
from blinkit_data  
group by Outlet_Location_Type  
order by Total_sales desc
```

	Outlet_Location_Type	Total_Sales
1	Tier 3	472133.03
2	Tier 2	393150.64
3	Tier 1	336397.81

11. All Metrics by Outlet Type:

```
select Outlet_Type,
       cast(sum(Total_Sales) as decimal(10,2)) as Total_Sales,
       cast(avg(Total_Sales) as decimal(10,0)) as Avg_Sales,
       count(*) as No_Of_Items,
       cast(avg(Rating) as decimal(10,2)) as Avg_Rating,
       cast(avg(Item_Visibility) as decimal(10,2)) as Item_visibility
  from blinkit_data
 group by Outlet_Type
 order by Total_Sales desc
```

	Outlet_Type	Total_Sales	Avg_Sales	No_Of_Items	Avg_Rating	Item_visibility
1	Supermarket Type1	787549.89	141	5577	3.96	0.06
2	Grocery Store	151939.15	140	1083	3.99	0.10
3	Supermarket Type2	131477.77	142	928	3.97	0.06
4	Supermarket Type3	130714.67	140	935	3.95	0.06

12. Top 5 items with the highest Sales

```
select top 5 Item_Identifier, sum(Total_Sales) as Total_Sales  
from blinkit_data  
group by Item_Identifier  
order by Total_Sales desc;
```

	Item_Identifier	Total_Sales
1	FDU12	2371.01119995117
2	FDT07	2306.89698791504
3	NCQ06	2294.71258544922
4	FDL58	2111.65438842773
5	NCB31	2104.72793579102

13. Outlet having the highest total Sales:

```
select top 1 Outlet_Identifier, sum(Total_Sales)as Total_Sales  
from blinkit_data  
group by Outlet_Identifier  
Order by Total_Sales desc;
```

	Outlet_Identifier	Total_Sales
1	OUT035	133103.907154083

14. Top 3 best selling item types in Tier 3 locations:

```
select top 3 Item_Type, sum(Total_Sales) as Total_Sales  
from blinkit_data  
where Outlet_Location_Type = 'Tier 3'  
group by Item_Type  
order by Total_Sales desc;
```

	Item_Type	Total_Sales
1	Fruits and Vegetables	70729.709312439
2	Snack Foods	68690.5912742615
3	Household	53724.1356124878

15. Outlet type that sells the most no. of distinct items:

```
select Outlet_Type, count(distinct Item_Identifier) as Distinct_Items  
from blinkit_data  
group by Outlet_Type  
order by Distinct_Items desc;
```

	Outlet_Type	Distinct_Items
1	Supermarket Type1	1550
2	Supermarket Type3	935
3	Supermarket Type2	928
4	Grocery Store	901

16. Showing yearly_Trend : average Sales per Establishment year of Outlet:

```
select Outlet_Establishment_Year, round(avg(Total_Sales), 2) as Avg_Sales  
from blinkit_data  
group by Outlet_Establishment_Year  
order by Outlet_Establishment_Year;
```

The screenshot shows a SQL query results window with two tabs: 'Results' and 'Messages'. The 'Results' tab is selected, displaying a table with two columns: 'Outlet_Establishment_Year' and 'Avg_Sales'. The data is as follows:

	Outlet_Establishment_Year	Avg_Sales
1	1998	139.8
2	2000	141.43
3	2010	142.06
4	2011	140.78
5	2012	140.3
6	2015	140.95
7	2017	143.12
8	2020	139.42
9	2022	141.68

17. Average Sales by Item fat content and Item type

```
select
```

```
    Item_Fat_Content,  
    Item_Type,  
    round(avg(Total_Sales),2) as Avg_Sales
```

```
from blinkit_data
```

```
group by Item_Fat_Content, Item_type
```

```
order by Avg_Sales desc;
```



	Item_Fat_Content	Item_Type	Avg_Sales
1	Regular	Seafood	180.67
2	Low Fat	Breakfast	165.26
3	Low Fat	Starchy Foods	157.97
4	Low Fat	Snack Foods	156.5
5	Regular	Fruits and Vegetables	152.98
6	Regular	Dairy	150.69
7	Low Fat	Household	149.42
8	Regular	Breads	148.99
9	Low Fat	Dairy	147.12
10	Regular	Meat	145.8
11	Regular	Canned	140
12	Low Fat	Canned	139.55
13	Low Fat	Frozen Foods	139.02
14	Regular	Frozen Foods	137.93
15	Low Fat	Hard Drinks	137.08
16	Low Fat	Fruits and Vegetables	136.56
17	Regular	Starchy Foods	135.25
18	Low Fat	Breads	134.58
19	Low Fat	Soft Drinks	133.34
20	Low Fat	Others	132.85
21	Regular	Baking Goods	132.42
22	Regular	Snack Foods	132.16
23	Low Fat	Meat	131
24	Low Fat	Health and Hygiene	130.82
25	Regular	Breakfast	127.84
26	Regular	Soft Drinks	121.77

18. The Outlet that sells the most variety of items:

```
select Outlet_Identifier,  
       count(distinct Item_Identifier)as unique_Items_Sold  
from blinkit_data  
group by Outlet_Identifier  
order by Unique_Items_Sold desc;
```

	Outlet_Identifier	unique_Items_Sold
1	OUT027	935
2	OUT013	932
3	OUT035	930
4	OUT046	930
5	OUT049	930
6	OUT045	929
7	OUT018	928
8	OUT017	926
9	OUT010	555
10	OUT019	528

19. The top 3 Outlet Types by Total Sales:

```
select top 3
    Outlet_Type,
    sum(Total_Sales) as Total_Sales
from blinkit_data
group by Outlet_Type
order by Total_Sales desc;
```

	Outlet_Type	Total_Sales
1	Supermarket Type1	787549.886491776
2	Grocery Store	151939.146755219
3	Supermarket Type2	131477.772266388

20. The heaviest items by average weight per item type:

```
select  
    Item_Type,  
        round(avg(Item_Weight),2) as Avg_Weight  
from blinkit_data  
group by Item_Type  
order by Avg_Weight desc;
```

	Item_Type	Avg_Weight
1	Others	13.85
2	Starchy Foods	13.69
3	Dairy	13.43
4	Household	13.38
5	Fruits and Vegetables	13.22
6	Health and Hygiene	13.14
7	Snack Foods	12.99
8	Frozen Foods	12.87
9	Meat	12.82
10	Breakfast	12.77
11	Seafood	12.55
12	Canned	12.31
13	Baking Goods	12.28
14	Soft Drinks	11.85
15	Hard Drinks	11.4
16	Breads	11.35

21. Items that are rarely visible

(visibility < 0.01) but have high sales

(top 10):

```
select top 10
    Item_Identifier,
    Item_Visibility,
    Total_Sales
from blinkit_data
where Item_Visibility < 0.01
order by Total_Sales desc;
```



	Item_Identifier	Item_Visibility	Total_Sales
1	FDK51	0.00876255612820387	266.688385009766
2	FDK51	0.0052432818338275	266.188385009766
3	FDK51	0.00523514300584793	265.888397216797
4	FDK51	0.00520979100838304	265.288391113281
5	FDK51	0.00523078581318259	265.188385009766
6	FDY02	0	264.790985107422
7	FDK51	0.00526475487276912	264.588409423828
8	FDK51	0.00523415300995111	264.088409423828
9	FDI15	0	263.788391113281
10	FDL58	0	263.756805419922

Conclusion:

From the analysis of the blinkit_data dataset, we conclude that the Item_Fat_Content column had multiple inconsistent representations for the same categories, such as 'LF', 'low fat', and 'reg'. After standardizing these values, we found that the data effectively falls into two main categories: 'Low Fat' and 'Regular'. This shows that most products are marketed with a focus on fat content, and a significant number of items belong to the 'Low Fat' category.

This conclusion suggests a trend or business strategy favoring low-fat products, possibly reflecting consumer health awareness or inventory patterns. The cleaned data is now more reliable for further analysis, such as nutritional profiling, customer segmentation, or sales forecasting.