

BLINK IT

GROCERY SALES ANALYSIS USING SQL



Presented by : Nidhi Singh

Bussiness Problem

BlinkIT (a grocery retail chain) manages multiple outlets across different city locations and sizes.

Each outlet sells hundreds of grocery items differing in type, fat content, visibility, and weight.

Despite large-scale operations, the company struggles to understand which factors truly impact sales and customer ratings.

SQL Questions:

1. Show all unique outlet types.
2. Find the total number of items sold from each outlet.
3. Find the average rating of each outlet type.
4. Display the top 10 items based on total sales.
5. Find the average sales of items based on fat content.
6. Show yearly sales based on the outlet establishment year.
7. Find total sales from each location type.
8. Find the average sales for each outlet size.
9. Show the top 5 item types by average rating.
10. Find the average item weight for each item type.

1. Show all unique outlet types.

Query

```
SELECT DISTINCT OutletType  
FROM items;
```

Result

	OutletType
▶	Supermarket Type 1
	Supermarket Type 2
	Grocery Store
	Supermarket Type 3

2. Find the total number of items sold from each outlet.

Query

```
SELECT OutletIdentifier, SUM(TotalSales) AS Total_Sales  
FROM items  
GROUP BY OutletIdentifier  
ORDER BY Total_Sales DESC;
```

Result

	OutletIdentifier	Total_Sales
▶	OUT035	133103.90699999999
	OUT046	132113.36980000007
	OUT013	131809.01560000007
	OUT018	131477.77639999994
	OUT045	130942.7802

3. Find the average rating of each outlet type.

Query

```
SELECT OutletType, ROUND(AVG(Rating), 2) AS Avg_Rating  
FROM items  
GROUP BY OutletType  
ORDER BY Avg_Rating DESC;
```

Result

	OutletType	Avg_Rating
▶	Grocery Store	3.98
	Supermarket Type1	3.95
	Supermarket Type2	3.95
	Supermarket Type3	3.95

4. Display the top 10 items based on total sales.

Query

```
SELECT ItemIdentifier, ItemType, SUM(TotalSales) AS Total_Revenue  
FROM items  
GROUP BY ItemIdentifier, ItemType  
ORDER BY Total_Revenue DESC  
LIMIT 10;
```

Result

	ItemIdentifier	ItemType	Total_Revenue
▶	FDU12	Baking Goods	2371.0112
	FDT07	Fruits and Vegetables	2306.897
	NCQ06	Household	2294.7126
	FDL58	Snack Foods	2111.6544
	NCB31	Household	2104.7279999999996
	FDX31	Fruits and Vegetables	2104.4622
	FDF05	Frozen Foods	2103.128
	FDR59	Breads	2096.5752
	FDP28	Frozen Foods	2087.8488
	FDA04	Frozen Foods	2072.0696

5. Find the average sales of items based on fat content.

Query

```
SELECT ItemFatContent, ROUND(AVG(TotalSales), 2) AS Avg_Sales  
FROM items  
GROUP BY ItemFatContent  
ORDER BY Avg_Sales DESC;
```

Result

	ItemFatContent	Avg_Sales
▶	Regular	141.5
	Low Fat	140.71

6. Show yearly sales based on the outlet establishment year.

Query

```
SELECT OutletEstablishmentYear, SUM(TotalSales) AS Yearly_Sales  
FROM items  
GROUP BY OutletEstablishmentYear  
ORDER BY OutletEstablishmentYear;
```

Result

	OutletEstablishmentYear	Yearly_Sales
▶	2011	78131.56659999998
	2012	130476.85979999998
	2014	131809.01560000007
	2015	130942.7802
	2016	132113.36980000007
	2017	133103.9069999999
	2018	204522.25700000025
	2020	129103.96039999987
	2022	131477.77639999994

7. Find total sales from each location type.

Query

```
SELECT OutletLocationType, SUM(TotalSales) AS Total_Revenue  
FROM items  
GROUP BY OutletLocationType  
ORDER BY Total_Revenue DESC;
```

Result

	OutletLocationType	Total_Revenue
▶	Tier 3	472133.03319999954
	Tier 2	393150.64759999956
	Tier 1	336397.81199999945

8. Find the average sales for each outlet size.

Query

```
SELECT OutletSize, ROUND(AVG(TotalSales), 2) AS Avg_Sales  
FROM items  
GROUP BY OutletSize  
ORDER BY Avg_Sales DESC;
```

Result

	OutletSize	Avg_Sales
▶	High	142.04
	Small	141.7
	Medium	139.88

9. Show the top 5 item types by average rating.

Query

```
SELECT ItemType, ROUND(AVG(Rating), 2) AS Avg_Rating  
FROM items  
GROUP BY ItemType  
ORDER BY Avg_Rating DESC  
LIMIT 5;
```

Result

	ItemType	Avg_Rating
▶	Household	4.00
	Meat	4.00
	Canned	3.99
	Baking Goods	3.98
	Health and Hygiene	3.97

10. Find the average item weight for each item type.

```
SELECT ItemType, ROUND(AVG(ItemWeight), 2) AS Avg_Weight  
FROM items  
GROUP BY ItemType  
ORDER BY Avg_Weight DESC;
```

Query

Result

ItemType	Avg_Weight
Starchy Foods	12.03
Others	11.23
Household	11.16
Dairy	11.14
Fruits and Vegetables	10.94
Health and Hygiene	10.87
Frozen Foods	10.79
Snack Foods	10.69
Breakfast	10.33
Canned	10.22
Meat	10.16
Baking Goods	10.16
Seafood	10
Soft Drinks	9.96
Hard Drinks	9.75
Breads	9.22

**THANK
YOU**