**Blinkit SQL Analysis – Presentation with Code & Outputs**

1. **Data Preparation**

-- Data Cleaning

UPDATE blinkit\_data

SET

ï»¿Item\_Fat\_Content = CASE

WHEN ï»¿Item\_Fat\_Content IN ('LF' , 'low fat') THEN 'Low Fat'

WHEN ï»¿Item\_Fat\_Content = 'reg' THEN 'Regular'

ELSE ï»¿Item\_Fat\_Content

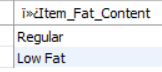
END;

select distinct(ï»¿Item\_Fat\_Content) from blinkitdb.blinkit\_data; #now we can see there is only two values in the column

**Output:**

* Fat content standardized → Only **Low Fat** & **Regular** remain.
* Ensures consistency for analysis.

**Visual:**



**--- Key Performing Indicators**

1. **Total Sales**

-- Finding KPI's as per business requirement

# Total Sales

# In SQL, CAST is a function used to convert one data type into another

SELECT

CONCAT(CAST(SUM(Sales)/1000000 AS DECIMAL(10,2)), ' M') AS Total\_Sales\_Millions

FROM

blinkitdb.blinkit\_data;

**Output Example:**

* **Total Sales = 1.20 M**

**Visual:**



1. **Average Sales & Items**

# Average Sales

select cast(avg(Sales) as decimal(10,2)) as 'Average\_Sales' from blinkitdb.blinkit\_data;

# Number of Items

select count(\*) as 'Number\_of\_Items' from blinkitdb.blinkit\_data;

**Output Example:**

* Average Sales = **₹141**
* Number of Items = **8523**

**Visual:**

1. **Average Rating**

SELECT CAST(AVG(Rating) AS DECIMAL(10,2)) AS avg\_rating FROM blinkit\_data;

**Output Example:**

* Avg Rating = ⭐ **3.97/ 5**

**Visual:**



1. **Sales by Fat Content**

SELECT Item\_Fat\_Content,

CONCAT(CAST(SUM(Sales)/1000000 AS DECIMAL(10,2)), ' M') AS Total\_Sales\_Millions,

CAST(AVG(Sales) AS DECIMAL(10,2)) AS Average\_Sales,

COUNT(\*) AS Number\_of\_Items,

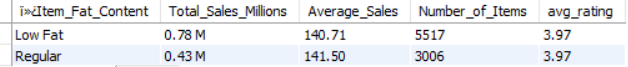
CAST(AVG(Rating) AS DECIMAL(10,2)) AS avg\_rating

FROM blinkit\_data

GROUP BY Item\_Fat\_Content

ORDER BY Total\_Sales\_Millions DESC;

**Output Example:**

****

1. **Sales by Item Type**

SELECT

Item\_Type, cast(SUM(Sales) as decimal(10,2)) AS Total\_Sales,

CAST(AVG(Sales) AS DECIMAL (10 , 2 )) AS 'Average\_Sales',

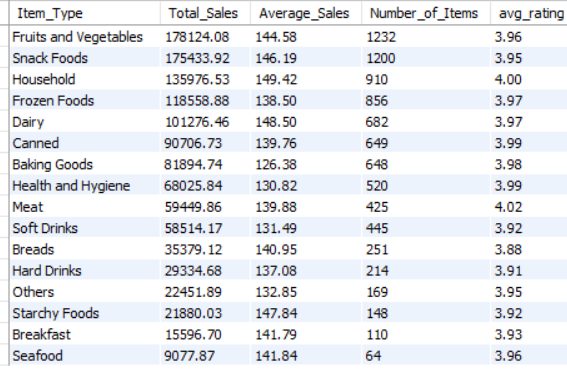
COUNT(\*) AS 'Number\_of\_Items',

CAST(AVG(Rating) AS DECIMAL (10 , 2 )) AS 'avg\_rating'

FROM

blinkitdb.blinkit\_data group by Item\_Type order by Total\_Sales desc;

**Output Example:**

****

1. **Fat Content by Outlet Location for Sales**

SELECT

Outlet\_Location\_Type,

ï»¿Item\_Fat\_Content,

CAST(SUM(Sales) AS DECIMAL (10 , 2 )) AS Total\_Sales,

CAST(AVG(Sales) AS DECIMAL (10 , 2 )) AS 'Average\_Sales',

COUNT(\*) AS 'Number\_of\_Items',

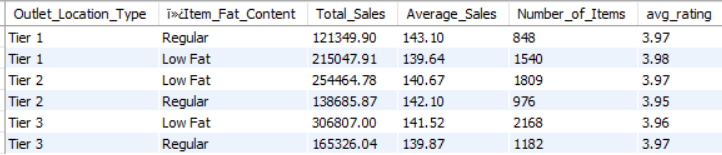
CAST(AVG(Rating) AS DECIMAL (10 , 2 )) AS 'avg\_rating'

FROM

blinkitdb.blinkit\_data

GROUP BY ï»¿Item\_Fat\_Content , Outlet\_Location\_Type

ORDER BY Outlet\_Location\_Type ASC;

**Output Example: **

1. **Sales by Outlet Establishment Year**

SELECT

Outlet\_Establishment\_Year,

CAST(SUM(Sales) AS DECIMAL (10 , 2 )) AS Total\_Sales,

CAST(AVG(Sales) AS DECIMAL (10 , 2 )) AS 'Average\_Sales',

COUNT(\*) AS 'Number\_of\_Items',

CAST(AVG(Rating) AS DECIMAL (10 , 2 )) AS 'avg\_rating'

FROM

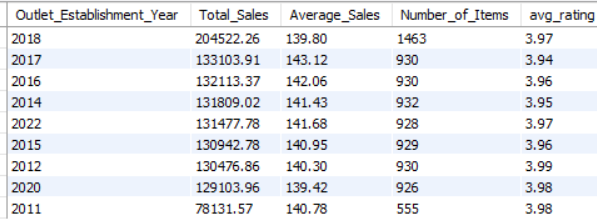
blinkitdb.blinkit\_data

GROUP BY Outlet\_Establishment\_Year

ORDER BY Total\_Sales DESC;

**Output Example:**

* Newer outlets (2010–2015) generate higher sales.
* Older outlets contribute steadily but less.



1. **Percentage of Sales by Outlet Size**

SELECT

Outlet\_Size,

CAST(SUM(Sales) AS DECIMAL(10,2)) AS Total\_Sales,

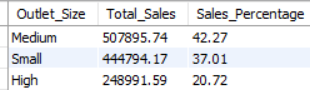
CAST((SUM(Sales) \* 100.0 / SUM(SUM(Sales)) OVER()) AS DECIMAL(10,2)) AS Sales\_Percentage

FROM blinkitdb.blinkit\_data

GROUP BY Outlet\_Size

ORDER BY Total\_Sales DESC;

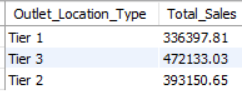
**Output Example:**



1. **Sales by Outlet Location**

select Outlet\_Location\_Type, cast( sum(Sales) as decimal(10,2)) as Total\_Sales from blinkitdb.blinkit\_data group by Outlet\_Location\_Type;

**Output Example:**



1. **All Metrics by Outlet Type**

SELECT

Outlet\_Type, cast(SUM(Sales) as decimal(10,2)) AS Total\_Sales,

CAST(AVG(Sales) AS DECIMAL (10 , 2 )) AS 'Average\_Sales',

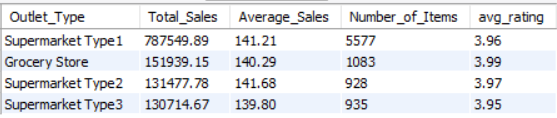
COUNT(\*) AS 'Number\_of\_Items',

CAST(AVG(Rating) AS DECIMAL (10 , 2 )) AS 'avg\_rating'

FROM

blinkitdb.blinkit\_data group by Outlet\_Type order by Total\_Sales desc;

**Output Example:**



**Key Insights:**

* Regular items slightly outperform than Low Fat.
* Fruits & Vegetables, Snacks, Household = top revenue categories.
* Tier 3 outlets sell more Regular items.
* Medium outlets dominate sales (42.27%).
* Supermarket Type 1 = highest performing outlet.

**Recommendations:**

1. Boost **top-selling categories** via promotions.
2. Strengthen **Tier 3 outlets** with Regular items.
3. Invest in **medium-size outlet expansion**.
4. Improve weaker outlets via targeted strategies.
5. Leverage **ratings** to improve customer trust.