**To change existing data**

UPDATE blinkit\_data

SET `Item Fat Content` =

CASE

WHEN `Item Fat Content` IN ('LF', 'low fat') THEN 'Low Fat'

WHEN `Item Fat Content` = 'reg' THEN 'Regular'

ELSE `Item Fat Content`

END;

select distinct(`Item Fat Content`) from blinkit\_data;

Output:

Regular

Low Fat

**KPI’s Requirments;**

**Total\_Sales**

SELECT CONCAT(CAST(SUM(Sales)/1000000 AS DECIMAL(10,2)), 'M') AS total\_sales\_millions

FROM blinkit\_data;



**Avg\_Sales**

SELECT CAST(avg(Sales) AS DECIMAL(10,2)) AS Avg\_sales

FROM blinkit\_data;



**No of Items**

SELECT count(\*) from blinkit\_data;

Output:8523



Granular Requirments:

Found out all those in one query (total sales,avg sales,no of items,avg rating for particular field like item fat content)

SELECT

`Item Fat Content`,

CAST(SUM(Sales) AS DECIMAL(10,2)) AS total\_sales,

CAST(AVG(Sales) AS DECIMAL(10,1)) AS Avg\_sales,

COUNT(\*) AS No\_of\_items,

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

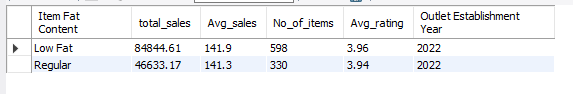
`Outlet Establishment Year`

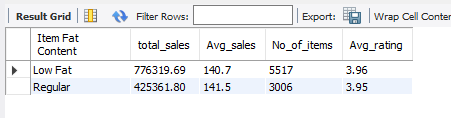
FROM blinkit\_data

WHERE `Outlet Establishment Year` = 2022

GROUP BY `Item Fat Content`, `Outlet Establishment Year`

ORDER BY total\_sales DESC;





Based on item tye:

SELECT

`Item Type`,

CAST(SUM(Sales) AS DECIMAL(10,2)) AS total\_sales,

CAST(AVG(Sales) AS DECIMAL(10,1)) AS Avg\_sales,

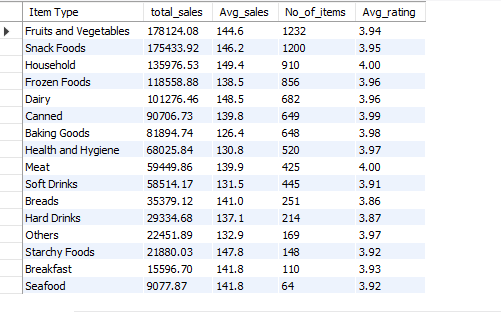
COUNT(\*) AS No\_of\_items,

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

FROM blinkit\_data

GROUP BY `Item Type`

ORDER BY total\_sales DESC;



**Fat Outlet loction with item fat content:**

**SELECT**

**`Outlet Location Type`,**

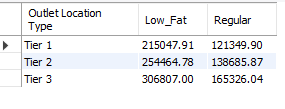
**IFNULL(cast(SUM(CASE WHEN `Item Fat Content` = 'Low Fat' THEN Sales ELSE 0 END)as decimal(10,2)), 0) AS Low\_Fat,**

**IFNULL(cast(SUM(CASE WHEN `Item Fat Content` = 'Regular' THEN Sales ELSE 0 END) as decimal(10,2)), 0) AS Regular**

**FROM blinkit\_data**

**GROUP BY `Outlet Location Type`**

**ORDER BY `Outlet Location Type`;**

****

**Total sales by outlet establishment:**

**SELECT `Outlet Establishment Year`,**

**cast(sum(Sales) as decimal(10,2)) as total\_sales,**

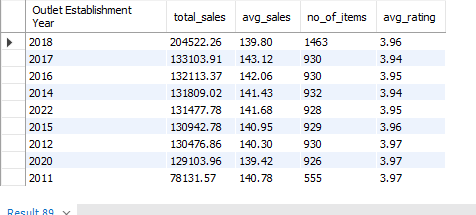
**cast(avg(Sales) as decimal(10,2)) as avg\_sales,**

**count(\*) as no\_of\_items,**

**cast(avg(Rating) as decimal(10,2)) as avg\_rating**

**from blinkit\_data**

**group by `Outlet Establishment Year` order by total\_sales desc;**

****

**Chart Requirments:**

**Percentage of sales by outet size:**

SELECT

`Outlet Size`,

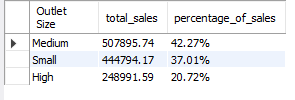
CAST(SUM(Sales) AS DECIMAL(10,2)) AS total\_sales,

CONCAT(ROUND(SUM(Sales) \* 100 / (SELECT SUM(Sales) FROM blinkit\_data), 2), '%') AS percentage\_of\_sales

FROM blinkit\_data

GROUP BY `Outlet Size`

ORDER BY total\_sales DESC;

****

**sales by outet location:**

SELECT

`Outlet Location Type`,

CAST(SUM(Sales) AS DECIMAL(10,2)) AS total\_sales,

cast(avg(Sales) as decimal(10,2)) as avg\_sales,

count(\*) as no\_of\_items,

cast(avg(Rating) as decimal(10,2)) as avg\_rating,

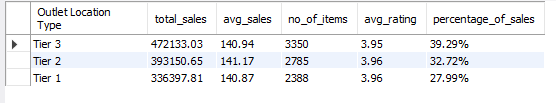
CONCAT(ROUND(SUM(Sales) \* 100 / (SELECT SUM(Sales) FROM blinkit\_data), 2), '%')

AS percentage\_of\_sales

FROM blinkit\_data

GROUP BY `Outlet Location Type`

ORDER BY total\_sales DESC;



**All Matrix by Out let type:**

SELECT

`Outlet Type`,

CAST(SUM(Sales) AS DECIMAL(10,2)) AS total\_sales,

cast(avg(Sales) as decimal(10,2)) as avg\_sales,

count(\*) as no\_of\_items,

cast(avg(Rating) as decimal(10,2)) as avg\_rating,

CONCAT(ROUND(SUM(Sales) \* 100 / (SELECT SUM(Sales) FROM blinkit\_data), 2), '%')

AS percentage\_of\_sales

FROM blinkit\_data

GROUP BY `Outlet Type`

ORDER BY total\_sales DESC;

