

Blinkit Sales Data Analysis

```
use Projects;  
select * from blinkit_data;
```

-- data cleaning

Item_fat_content have values LF, low fat, reg so these values should be updated to corresponding values.

```
select distinct item_fat_content from blinkit_data;
```

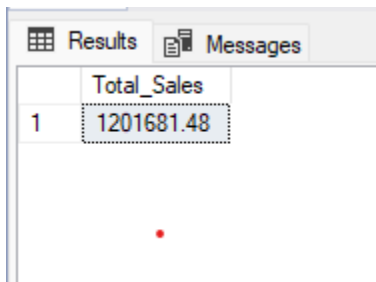
```
update blinkit_data  
set item_fat_content = case  
when item_fat_content in ('LF','low fat') then 'Low Fat'  
when item_fat_content = 'reg' then 'Regular'  
else  
item_fat_content  
end;
```

```
select distinct item_fat_content from blinkit_data;
```

-- KPI's Requirement

-- Total Sales

```
select cast(sum(Total_sales) as decimal(10,2)) as Total_Sales  
from blinkit_data;
```

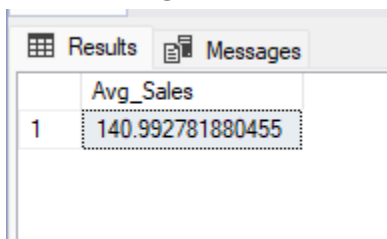


A screenshot of a SQL query results window. The window has two tabs: 'Results' and 'Messages'. The 'Results' tab is active, showing a table with one column 'Total_Sales' and one row with the value '1201681.48'. The table is highlighted with a blue border.

	Total_Sales
1	1201681.48

-- Overall Average Sales

```
select avg(total_sales) as Avg_Sales from blinkit_data;
```



A screenshot of a SQL query results window. The window has two tabs: 'Results' and 'Messages'. The 'Results' tab is active, showing a table with one column 'Avg_Sales' and one row with the value '140.992781880455'. The table is highlighted with a blue border.

	Avg_Sales
1	140.992781880455

-- Average Sales for each Unique Items

```
select cast(sum(total_sales) / count(distinct item_identifier) as decimal(10,2)) as  
Average_Sales_Per_Item from blinkit_data;
```

Results		Messages
	Average_Sales_Per_Item	
1	770.80	

-- Number of items

```
select count(item_identifier) as no_of_items  
from blinkit_data;
```

Results		Messages
	no_of_items	
1	8523	

-- Number of Unique items

```
select count(distinct item_identifier) as no_of_items  
from blinkit_data;
```

Results		Messages
	no_of_items	
1	1559	

-- Overall Average rating

```
select cast(avg(rating) as decimal(10,2)) As Average_Rating from blinkit_data;
```

Results		Messages
	Average_Rating	
1	3.97	

-- Average rating for each unique item

```
select distinct item_identifier, avg(Rating) as Avg_Rating
from blinkit_data
group by item_identifier
order by Avg_Rating desc;
```

Results		Messages
	item_identifier	Avg_Rating
1	DRF48	5
2	FDR03	4.75
3	FDU47	4.666666666666667
4	NCV18	4.65000009536743
5	NCR54	4.58000001907349
6	FDW20	4.54999995231628
7	FDK32	4.53333346048991
8	FDI12	4.52500003576279
9	FDE20	4.52000007629395
10	FDG35	4.51999998092651
11	DRJ13	4.51666673024495
12	FDU58	4.5
13	DRI59	4.5
14	FDI14	4.48571423121861
15	NCN42	4.48000001907349
16	NCT53	4.46666669845581
17	FDP16	4.45999994277954
18	FDX10	4.44999998807907
19	NCT05	4.44999992847443
20	FDG07	4.44000000000000

Query executed successfully.

-- Total Sales,Average Sales, no of items, Avg_Rating by Fat Content

```
select item_fat_content, sum(total_sales) as Total_sales, avg(total_sales) As Avg_Sales,
count(item_identifier) as No_of_Items, avg(rating) as Avg_Rating from blinkit_data
group by item_fat_content;
```

Results

Messages

	item_fat_content	Total_sales	Avg_Sales	No_of_Items	Avg_Rating
1	Low Fat	776319.67764473	140.714097814887	5517	3.96628602367392
2	Regular	425361.802322388	141.50425892295	3006	3.96506985841556

-- Total Sales,Average Sales, no of items, Avg_Rating by Item Type

```
select item_type, cast(sum(total_sales) as decimal(10,2)) as Total_Sales,
cast(avg(total_sales) as decimal(10,1)) as Avg_Sales, cast(count(item_identifier) as
decimal(10,2)) as no_of_items,cast(avg(rating) as decimal(10,2)) as Avg_Rating
from blinkit_data group by item_type;
```

	item_type	Total_Sales	Avg_Sales	no_of_items	Avg_Rating
1	Snack Foods	175433.92	146.2	1200.00	3.95
2	Seafood	9077.87	141.8	64.00	3.96
3	Breads	35379.12	141.0	251.00	3.88
4	Canned	90706.73	139.8	649.00	3.99
5	Dairy	101276.46	148.5	682.00	3.97
6	Baking Goods	81894.74	126.4	648.00	3.98
7	Others	22451.89	132.9	169.00	3.95
8	Breakfast	15596.70	141.8	110.00	3.93
9	Fruits and Vegetables	178124.08	144.6	1232.00	3.96
10	Frozen Foods	118558.88	138.5	856.00	3.97
11	Health and Hygiene	68025.84	130.8	520.00	3.99
12	Meat	59449.86	139.9	425.00	4.02
13	Starchy Foods	21880.03	147.8	148.00	3.92
14	Soft Drinks	58514.16	131.5	445.00	3.92
15	Hard Drinks	29334.68	137.1	214.00	3.91
16	Household	135976.53	149.4	910.00	4.00

-- Fat Content by outlet for Total Sales & Pivot the Data

```
select outlet_location_type, item_fat_content,cast(sum(total_sales) as decimal(10,2)) as
Total_Sales,cast(avg(total_sales) as decimal(10,1)) as Avg_Sales,
cast(count(item_identifier) as decimal(10,2)) as no_of_items,
cast(avg(rating) as decimal(10,2)) as Avg_Rating from blinkit_data
group by outlet_location_type, item_fat_content
order by item_fat_content, outlet_location_type;
```

	outlet_location_type	item_fat_content	Total_Sales	Avg_Sales	no_of_items	Avg_Rating
1	Tier 1	Low Fat	215047.91	139.6	1540.00	3.98
2	Tier 2	Low Fat	254464.77	140.7	1809.00	3.97
3	Tier 3	Low Fat	306806.99	141.5	2168.00	3.96
4	Tier 1	Regular	121349.90	143.1	848.00	3.97
5	Tier 2	Regular	138685.87	142.1	976.00	3.95
6	Tier 3	Regular	165326.03	139.9	1182.00	3.97

```
CREATE TABLE backup_outlet_fat (
    outlet_location_type VARCHAR(255),
    item_fat_content VARCHAR(255),
    Total_Sales DECIMAL(10, 2),
    Avg_Sales DECIMAL(10, 1),
    no_of_items DECIMAL(10, 2),
    Avg_Rating DECIMAL(10, 2)
);
```

```
-- insert into backup_outlet_fat
```

```
insert into backup_outlet_fat
```

```
select outlet_location_type, item_fat_content, cast(sum(total_sales) as decimal(10,2)) as
Total_Sales, cast(avg(total_sales) as decimal(10,1)) as Avg_Sales,
cast(count(item_idenfier) as decimal(10,2)) as no_of_items,
cast(avg(rating) as decimal(10,2)) as Avg_Rating from blinkit_data
group by outlet_location_type, item_fat_content
order by item_fat_content, outlet_location_type;
```

```
select * from backup_outlet_fat;
```

```
select outlet_location_type,
sum(case when item_fat_content = 'Low Fat' then total_sales else 0 end) as
Low_Fat_Total_Sales,
avg(case when item_fat_content = 'Low Fat' then Avg_sales else 0 end) as
Low_Fat_Avg_Sales,
sum(case when item_fat_content = 'Low Fat' then no_of_items else 0 end) as
Low_Fat_Count_items,
avg(case when item_fat_content = 'Low Fat' then Avg_rating else 0 end) as
Low_Fat_avg_rating,
```

```
sum(case when item_fat_content = 'Regular' then total_sales else 0 end) as
Reg_Total_Sales,
avg(case when item_fat_content = 'Regular' then Avg_sales else 0 end) as
Reg_Avg_Sales,
sum(case when item_fat_content = 'Regular' then no_of_items else 0 end) as
Reg_Count_items,
avg(case when item_fat_content = 'Regular' then Avg_rating else 0 end) as
Reg_avg_rating from backup_outlet_fat group by outlet_location_type;
```

Results		Messages							
	outlet_location_type	Low_Fat_Total_Sales	Low_Fat_Avg_Sales	Low_Fat_Count_items	Low_Fat_avg_rating	Reg_Total_Sales	Reg_Avg_Sales	Reg_Count_items	Reg_avg_rating
1	Tier 1	215047.91	69.800000	1540.00	1.990000	121349.90	71.550000	848.00	1.985000
2	Tier 2	254464.77	70.350000	1809.00	1.985000	138685.87	71.050000	976.00	1.975000
3	Tier 3	306806.99	70.750000	2168.00*	1.980000	165326.03	69.950000	1182.00	1.985000

Total Sales by Outlet Establishment

```
select outlet_establishment_year, sum(total_sales) as Total_sales
from blinkit_data
group by outlet_establishment_year;
```

Results Messages		
	outlet_establishment_year	Total_sales
1	1998	204522.256774902
2	2010	132113.369853973
3	2022	131477.772266388
4	2000	131809.015523911
5	2020	129103.956199646
6	2011	78131.5644340515
7	2017	133103.907154083
8	2012	130476.859680176
9	2015	130942.778079987

-- Percentage of sales by outlet size

```
select outlet_size, sum(total_sales) as Total_Sales, cast(sum(total_sales) * 100 / (select
sum(total_sales) from blinkit_data) as decimal(10,2)) as PerSales_Outlet from blinkit_data
group by outlet_size;
```

Results Messages			
	outlet_size	Total_Sales	PerSales_Outlet
1	High	248991.583692551	20.72
2	Medium	507895.72769165	42.27
3	Small	444794.168582916	37.01

-- Sales by outlet location

```
select outlet_identifier, sum(total_sales) as Total_Sales,
cast(sum(total_sales) * 100 / (select sum(total_sales) from blinkit_data) as decimal(10,2))
as PerSales_Outlet from blinkit_data group by outlet_identifier
order by PerSales_outlet desc;
```

Results		Messages	
	outlet_identifier	Total_Sales	PerSales_Outlet
1	OUT035	133103.907154083	11.08
2	OUT046	132113.369853973	10.99
3	OUT013	131809.015523911	10.97
4	OUT018	131477.772266388	10.94
5	OUT045	130942.778079987	10.90
6	OUT027	130714.674453735	10.88
7	OUT049	130476.859680176	10.86
8	OUT017	129103.956199646	10.74
9	OUT010	78131.5644340515	6.50
10	OUT019	73807.582321167	6.14

-- All metrics by outlet type

```
select outlet_type, sum(total_sales) as Total_Sales,
cast(sum(total_sales) * 100 / (select sum(total_sales) from blinkit_data) as decimal(10,2))
as PerSales_Outlet,
avg(Total_sales) as Avg_Sales, count(item_identifier) as no_of_items, avg(rating) as
Avg_Rating from blinkit_data group by outlet_type order by outlet_type;
```

Results Messages

	outlet_type	Total_Sales	PerSales_Outlet	Avg_Sales	no_of_items	Avg_Rating
1	Grocery Store	151939.146755219	12.64	140.294687677949	1083	3.98587257732205
2	Supermarket Type1	787549.886491776	65.54	141.213893937919	5577	3.96324188388204
3	Supermarket Type2	131477.772266388	10.94	141.678633907746	928	3.9712284484814
4	Supermarket Type3	130714.674453735	10.88	139.80179085961	935	3.95294117557811