SQL Project : Zepto

Step 1: Getting familiar with the dataset

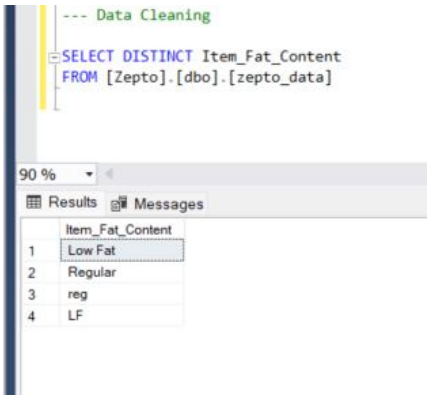
such as – number of columns, datatype, columns names, number of records, etc

SELECT * FROM [Zepto].[dbo].[zepto_data]

SELECT COUNT(*) FROM [Zepto].dbo.zepto_data

Step 2: Data Cleaning

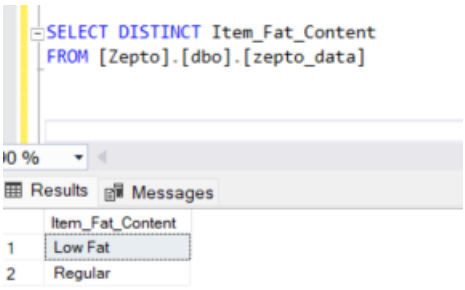
We can see in below image that the field Item_Fat_content has multiple variations of the same category (e.g., LF, Low Fat) that can cause issues in reporting, aggregations, and filtering.



We need to clean this field to ensure data consistency and accuracy in analysis. By standardizing these values, we can improve data quality, making it easier to generate insights and maintain uniformity in our datasets.

```
UPDATE [zepto_data]
SET Item_Fat_Content =
CASE WHEN [Item_Fat_Content] = 'LF' THEN 'Low Fat'
        WHEN [Item_Fat_Content] = 'reg' THEN 'Regular'
         ELSE [Item_Fat_Content]
         END
```

After firing this above query, as a result we get below results:

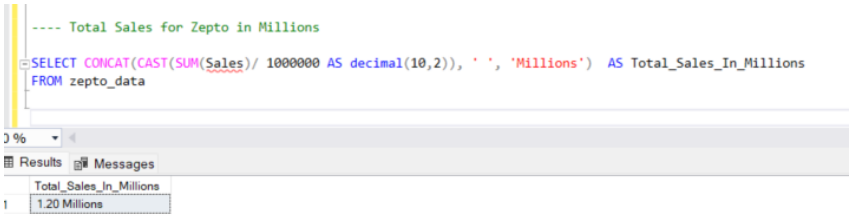


Step 3: Calculating KPIs

Total Sales

(in Millions)

```
SELECT CONCAT(CAST(SUM(Sales)/ 1000000 AS decimal(10,2)), ' ', 'Millions')  AS
Total_Sales_In_Millions
FROM zepto_data
```

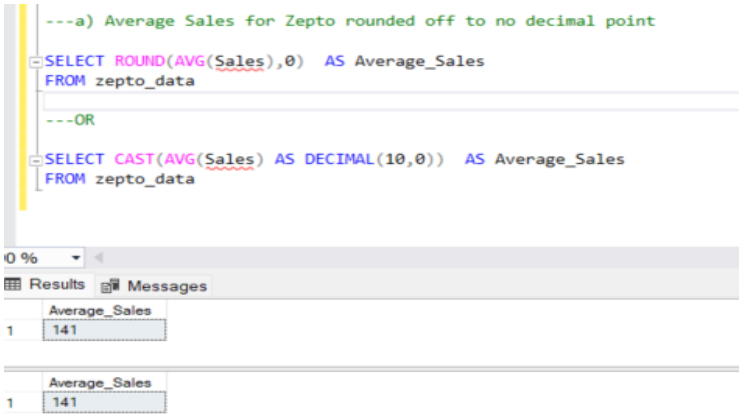


.

Average Sales

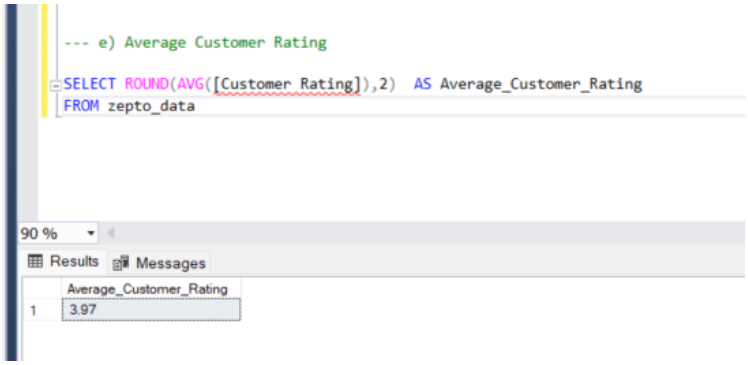
(rounded off to zero decimal point)

```
SELECT ROUND(AVG(Sales),0) AS Average_Sales
FROM zepto_data

---OR

SELECT CAST(AVG(Sales) AS DECIMAL(10,0))  AS Average_Sales
FROM zepto_data
```



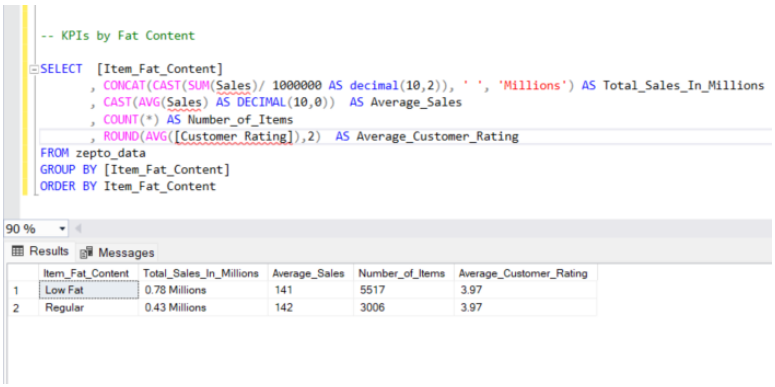
Average Customer Rating

```
SELECT ROUND(AVG([Customer Rating]),2)  AS Average_Customer_Rating
FROM zepto_data
```



KPIs by Fat Content

```
SELECT  [Item_Fat_Content]
        , CONCAT(CAST(SUM(Sales)/ 1000000 AS decimal(10,2)), ' ', 'Millions') AS Total_Sales_In_Millions
        , CAST(AVG(Sales) AS DECIMAL(10,0))  AS Average_Sales
        , COUNT(*) AS Number_of_Items
        , ROUND(AVG([Customer Rating]),2)  AS Average_Customer_Rating
FROM zepto_data
GROUP BY [Item_Fat_Content]
ORDER BY Item_Fat_Content
```



Top 5 selling items for Zepto

```
SELECT [Item_Type] Top_5_selling_items
from (
SELECT TOP 5 [Item_Type]
FROM zepto_data
GROUP BY [Item_Type]
ORDER BY CONCAT(CAST(SUM(Sales)/ 1000000 AS decimal(10,2)), ' ', 'Millions')
) p
ORDER BY [Item_Type]
ORDER BY Item_Type
```

```sql
---- Top 5 selling items

SELECT [Item_Type] Top_5_selling_items
from (
SELECT TOP 5 [Item_Type]
FROM zepto_data
GROUP BY [Item_Type]
ORDER BY CONCAT(CAST(SUM(Sales)/ 1000000 AS decimal(10,2)), ' ', 'Millions')
) p
ORDER BY [Item_Type]
```

90 %

Results | Messages

	Top_5_selling_items
1	Breakfast
2	Hard Drinks
3	Others
4	Seafood
5	Starchy Foods

Fat Content by Outlet for Total Sales

```sql
SELECT Outlet_Location_Type,
      ISNULL([Low Fat], 0) AS Low_Fat_Sales_In_Millions,
      ISNULL([Regular], 0) AS Regular_Sales_In_Millions
FROM
(
   SELECT Outlet_Location_Type, Item_Fat_Content,
        CAST(SUM(Sales)/ 1000000 AS decimal(10,2)) AS Total_Sales_Millions
   FROM zepto_data
   GROUP BY Outlet_Location_Type, Item_Fat_Content
) AS t1
PIVOT
(
   SUM(Total_Sales_Millions)
   FOR Item_Fat_Content IN ([Low Fat], [Regular])
) AS PivotTable
ORDER BY Outlet_Location_Type;
```

```sql
-- Fat Content by Outlet for Total Sales


SELECT Outlet_Location_Type,
       ISNULL([Low Fat], 0) AS Low_Fat_Sales_In_Millions,
       ISNULL([Regular], 0) AS Regular_Sales_In_Millions
FROM
(
    SELECT Outlet_Location_Type, Item_Fat_Content,
         CAST(SUM(Sales)/ 1000000 AS decimal(10,2)) AS Total_Sales_Millions
    FROM zepto_data
    GROUP BY Outlet_Location_Type, Item_Fat_Content
) AS t1
PIVOT
(
    SUM(Total_Sales_Millions)
    FOR Item_Fat_Content IN ([Low Fat], [Regular])
) AS PivotTable
ORDER BY Outlet_Location_Type;
```

90 %

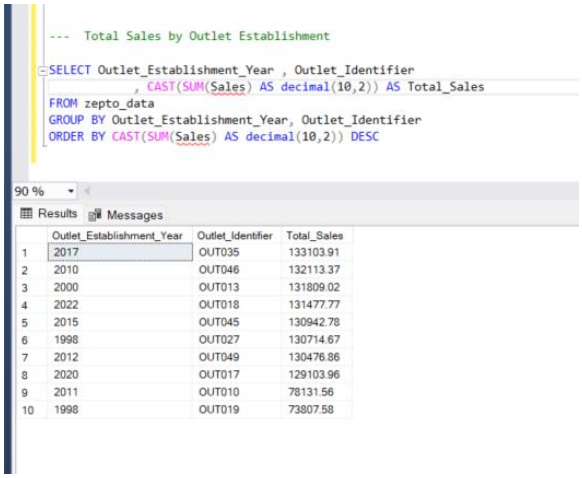
Results | Messages

	Outlet_Location_Type	Low_Fat_Sales_In_Millions	Regular_Sales_In_Millions
1	Tier 1	0.22	0.12
2	Tier 2	0.25	0.14
3	Tier 3	0.31	0.17

This query aims to transform the Zepto_data table to display total sales (in millions) for each combination of Outlet_Location_Type and Item_Fat_Content. The result will show Outlet_Location_Type as rows and Item_Fat_Content categories ("Low Fat" and "Regular") as columns. If there are no sales for a particular combination, the query will display 0 instead of NULL.

Total Sales by Outlet Establishment

```
SELECT Outlet_Establishment_Year , Outlet_Identifier
        , CAST(SUM(Sales) AS decimal(10,2)) AS Total_Sales
FROM zepto_data
GROUP BY Outlet_Establishment_Year, Outlet_Identifier
ORDER BY CAST(SUM(Sales) AS decimal(10,2)) DESC
```



```
--- Total Sales by Outlet Establishment

SELECT Outlet_Establishment_Year , Outlet_Identifier
        , CAST(SUM(Sales) AS decimal(10,2)) AS Total_Sales
FROM zepto_data
GROUP BY Outlet_Establishment_Year, Outlet_Identifier
ORDER BY CAST(SUM(Sales) AS decimal(10,2)) DESC
```

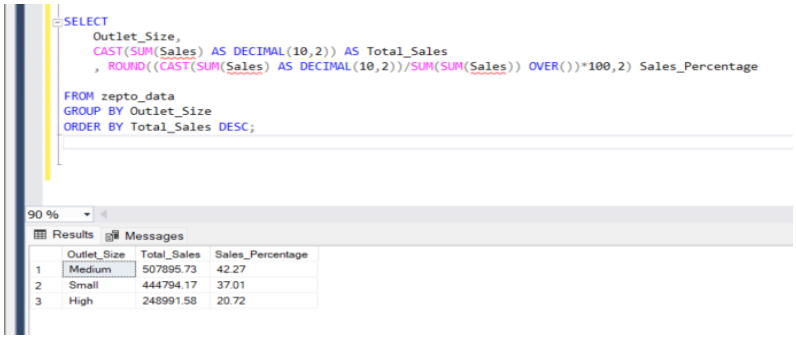
90 %

Results | Messages

	Outlet_Establishment_Year	Outlet_Identifier	Total_Sales
1	2017	OUT035	133103.91
2	2010	OUT046	132113.37
3	2000	OUT013	131809.02
4	2022	OUT018	131477.77
5	2015	OUT045	130942.78
6	1998	OUT027	130714.67
7	2012	OUT049	130476.86
8	2020	OUT017	129103.96
9	2011	OUT010	78131.56
10	1998	OUT019	73807.58

Percentage of Sales by Outlet Size

```
SELECT
    Outlet_Size,
    CAST(SUM(Sales) AS DECIMAL(10,2)) AS Total_Sales
     , ROUND((CAST(SUM(Sales) AS DECIMAL(10,2))/SUM(SUM(Sales)) OVER())*100,2) Sales_Percentage

FROM zepto_data
GROUP BY Outlet_Size
ORDER BY Total_Sales DESC;
```



```
SELECT
    Outlet_Size,
    CAST(SUM(Sales) AS DECIMAL(10,2)) AS Total_Sales
    , ROUND((CAST(SUM(Sales) AS DECIMAL(10,2))/SUM(SUM(Sales)) OVER())*100,2) Sales_Percentage

FROM zepto_data
GROUP BY Outlet_Size
ORDER BY Total_Sales DESC;
```

90 %

Results | Messages

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

SUM(Total_Sales): Within the GROUP BY context, this computes the total sales for each Outlet_Size.
SUM(…) OVER(): The outer SUM combined with the OVER() clause calculates the grand total of all Total_Sales across all outlet sizes without collapsing the result set.