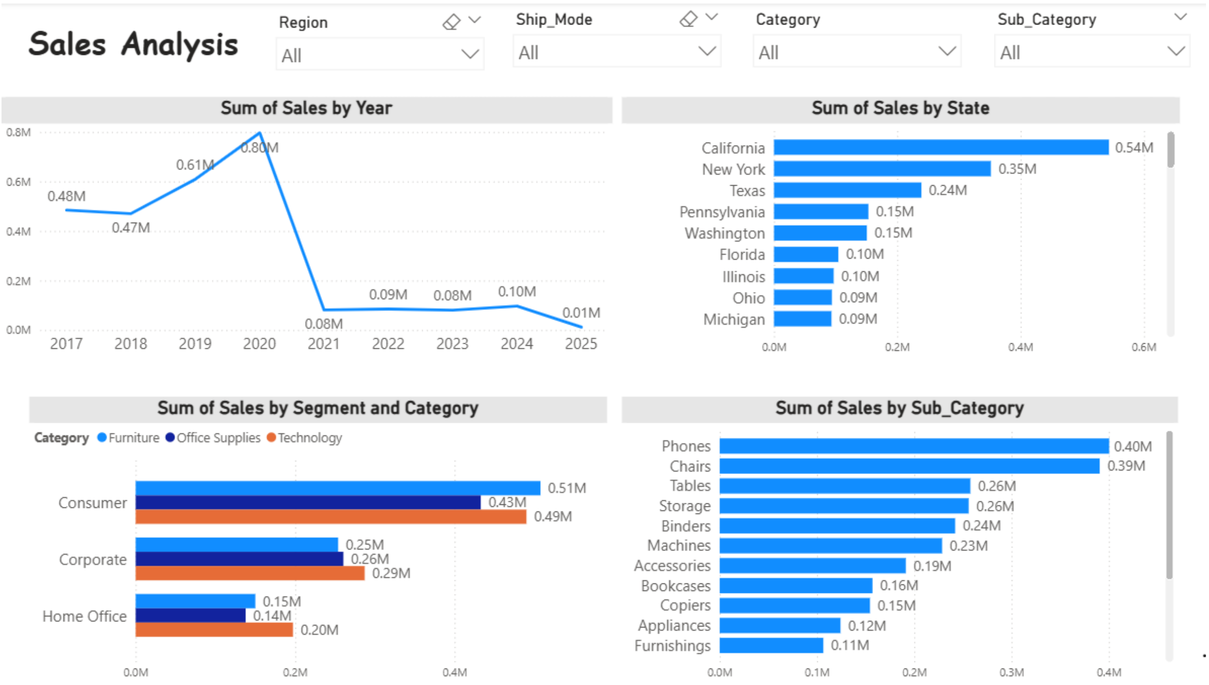
Dashboard & Reporting

Sales Dashboard



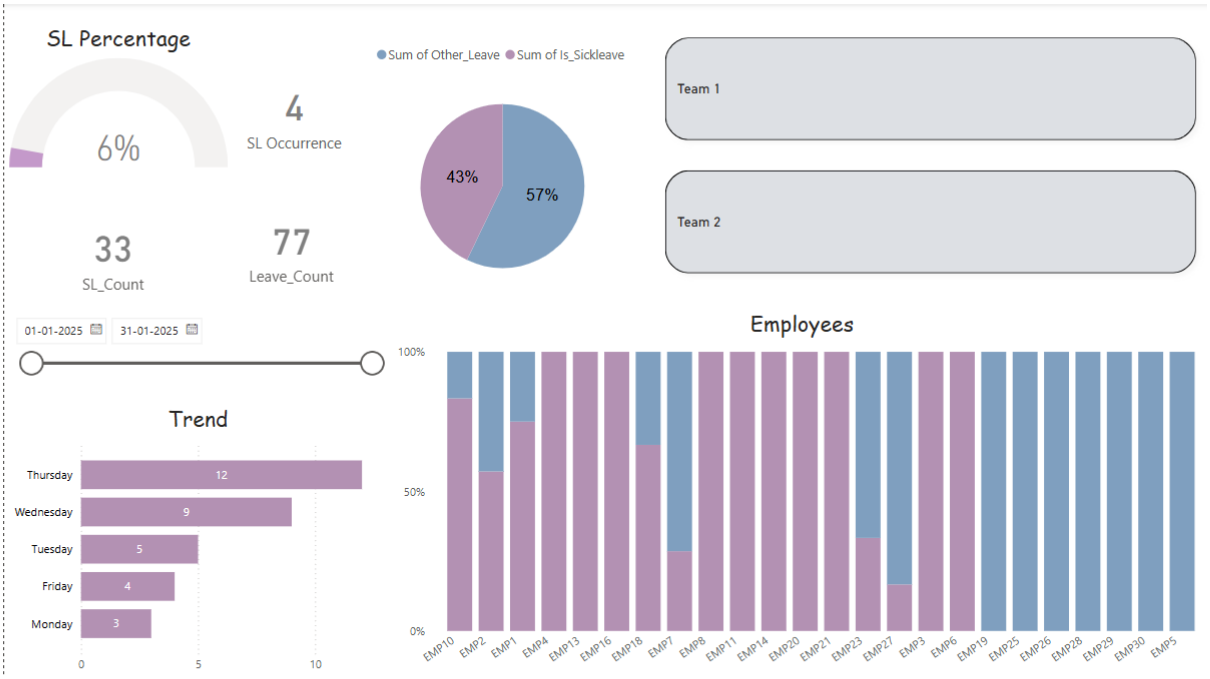
Region: All

Ship_Mode: All

Category: All

Sub_Category: All

Contact Center Reporting Dashboard

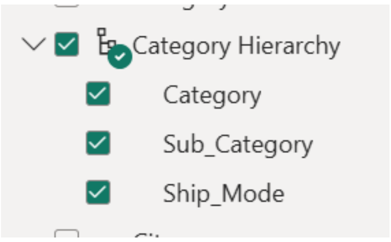


Using the same template, we can replicate productivity tracking to monitor utilization and key performance indicators (KPIs). Additionally, we can extend this approach to Quality Assessment (QA) and Training Need Analysis for a QA & Training Dashboard.

Dashboard & Reporting

Why Use a Hierarchy?

Drill-down capability, Reduces clutter, Saves space on reports, Improves user experience

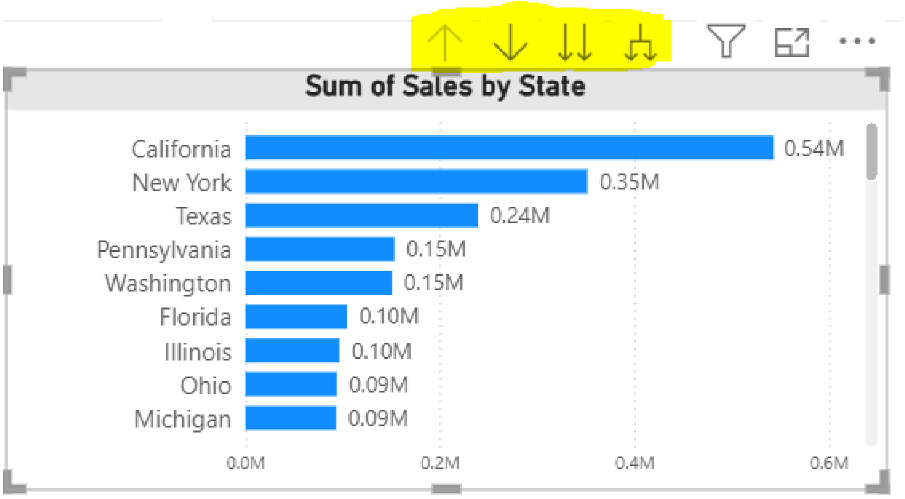


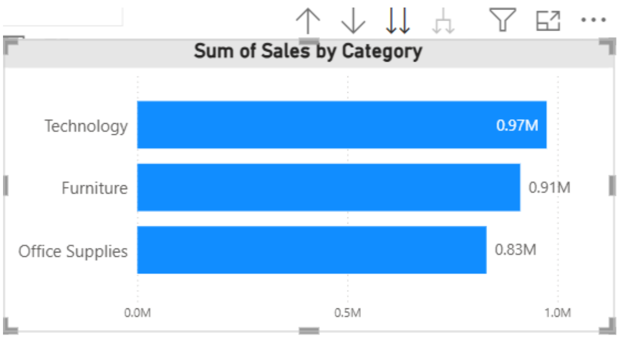
☑ Category Hierarchy

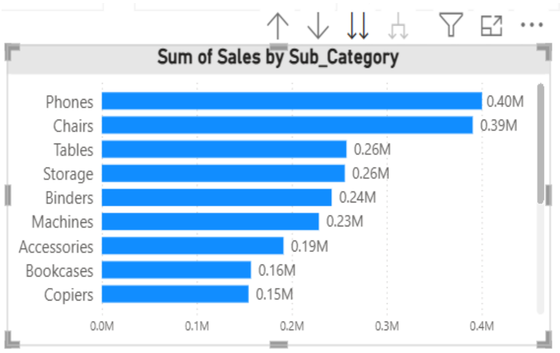
☑ Category

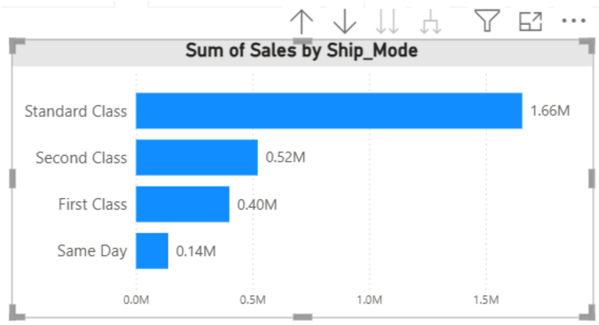
☑ Sub_Category

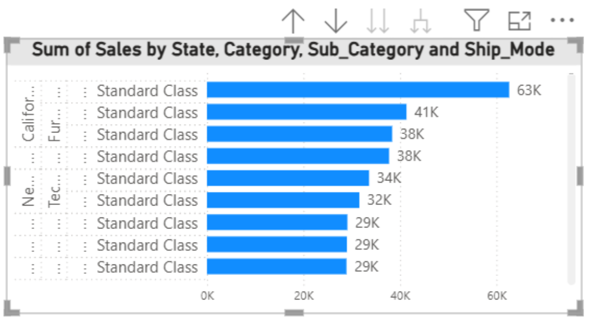
☑ Ship_Mode





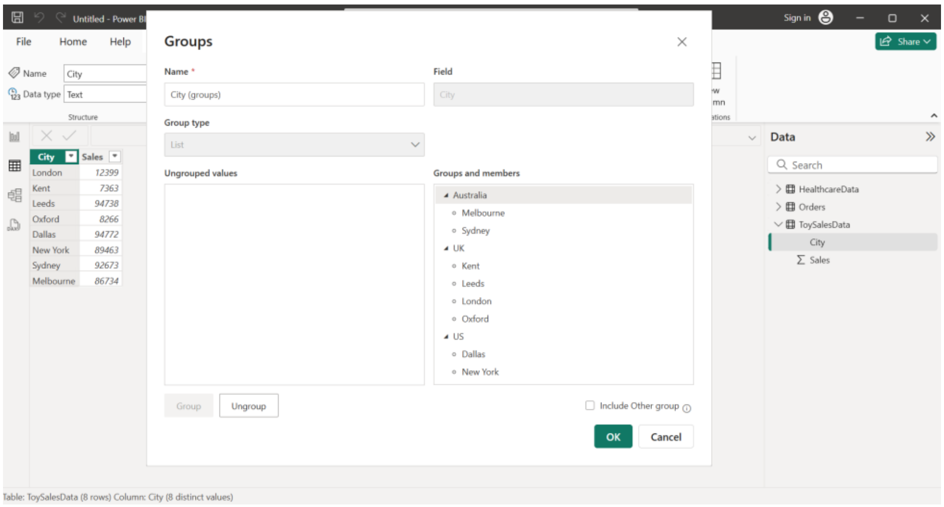






Dashboard & Reporting

New Group



Name: City
Data type: Text

City	Sales
London	12399
Kent	7363
Leeds	94738
Oxford	8266
Dallas	94772
New York	89463
Sydney	92673
Melbourne	86734

Groups

Name *: City (groups)

Field: City

Group type: List

Ungrouped values

Groups and members

- Australia
  - Melbourne
  - Sydney
- UK
  - Kent
  - Leeds
  - London
  - Oxford
- US
  - Dallas
  - New York

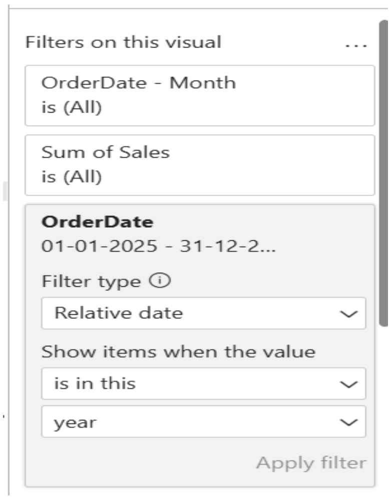
Group | Ungroup

□ Include Other group

OK | Cancel

Table: ToySalesData (8 rows) Column: City (8 distinct values)

Visual Level Filters:



Filters on this visual ...

OrderDate - Month
is (All)

Sum of Sales
is (All)

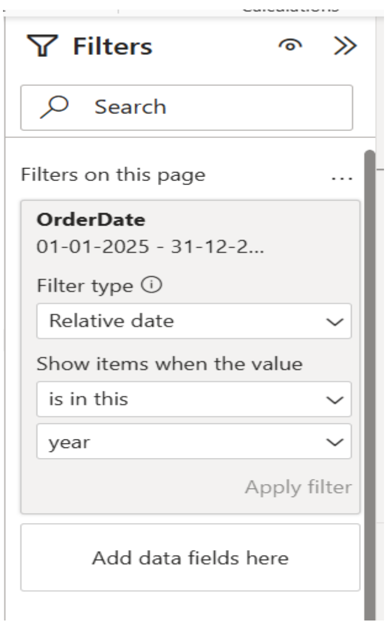
OrderDate
01-01-2025 - 31-12-2...

Filter type: Relative date

Show items when the value: is in this year

Apply filter

Page Level Filter:



Filters on this page ...

OrderDate
01-01-2025 - 31-12-2...

Filter type: Relative date

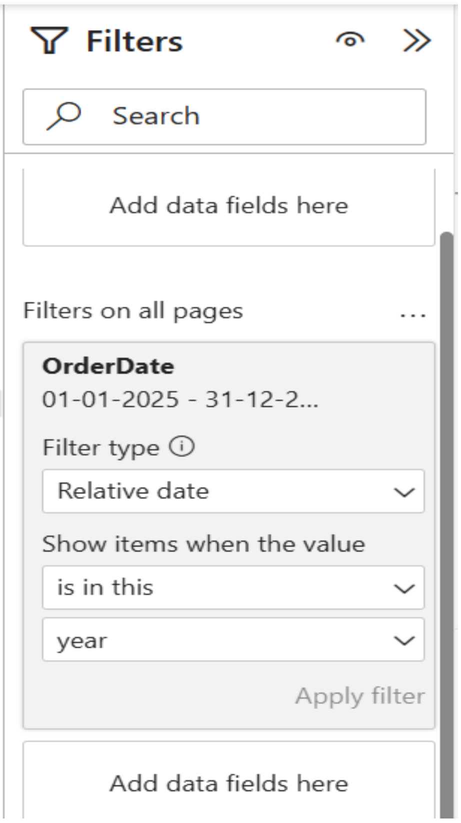
Show items when the value: is in this year

Apply filter

Add data fields here

Dashboard & Reporting

Page Level Filters:



Filters

Search

Add data fields here

Filters on all pages ...

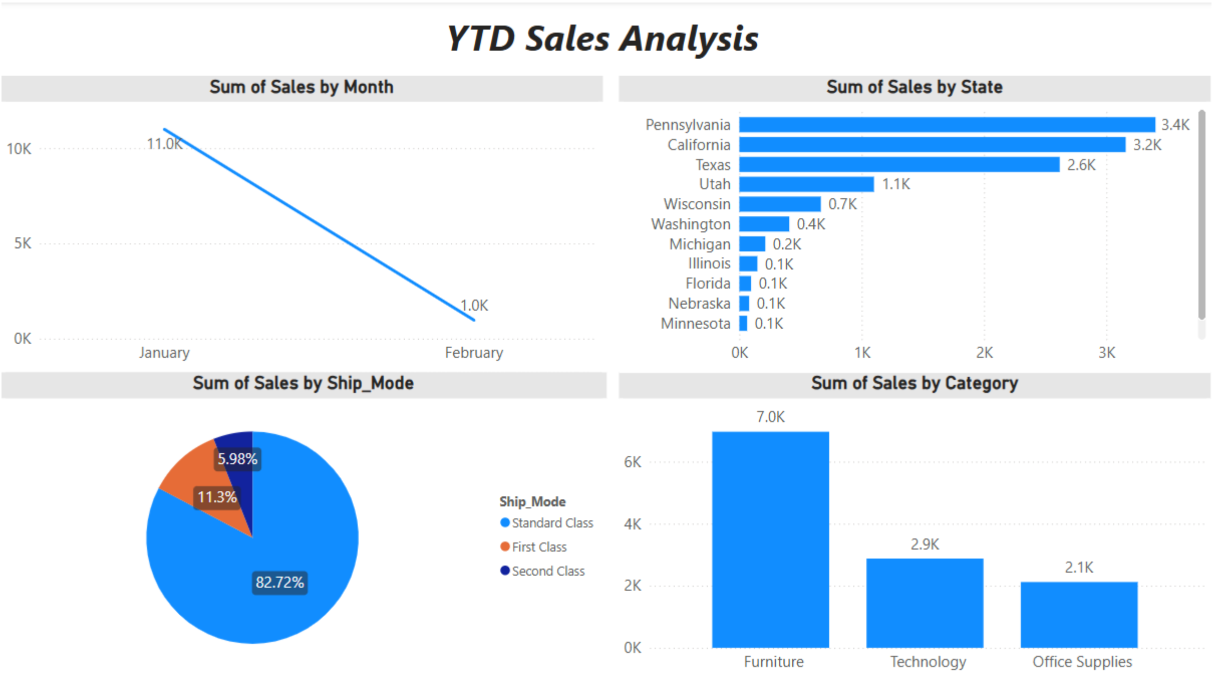
OrderDate
01-01-2025 - 31-12-2...

Filter type: Relative date

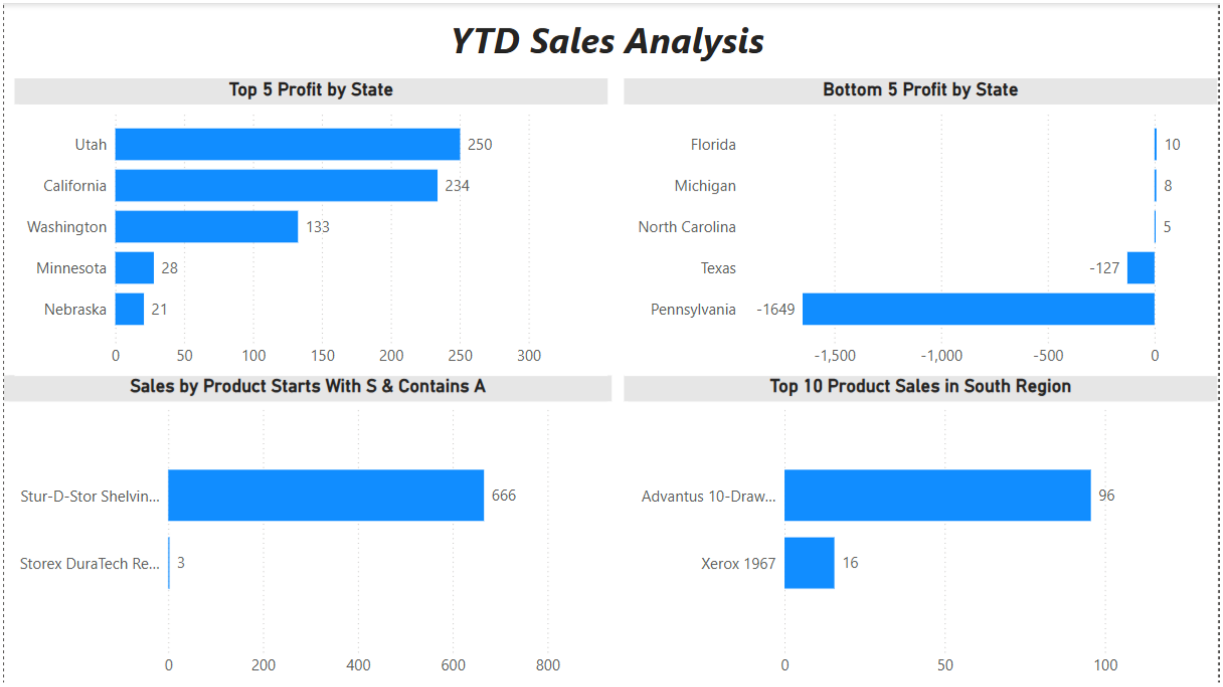
Show items when the value: is in this year

Apply filter

Add data fields here

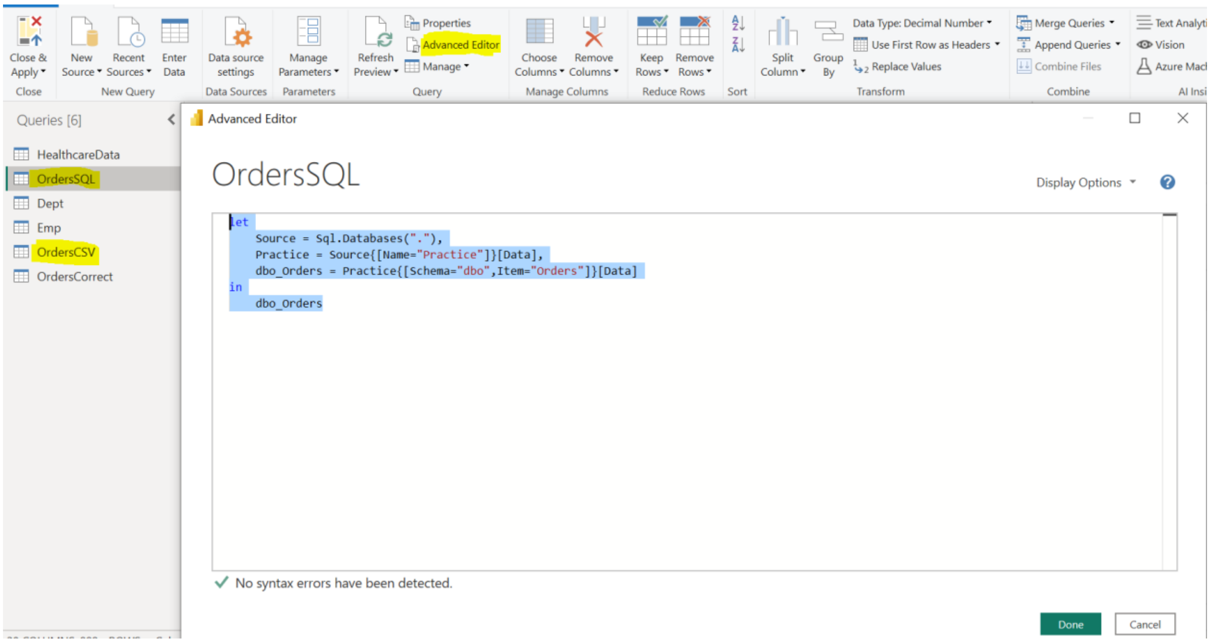


Dashboard & Reporting



Change Source upon UAT

Transform Data > Advanced Editor > Copy & paste the code of the New source

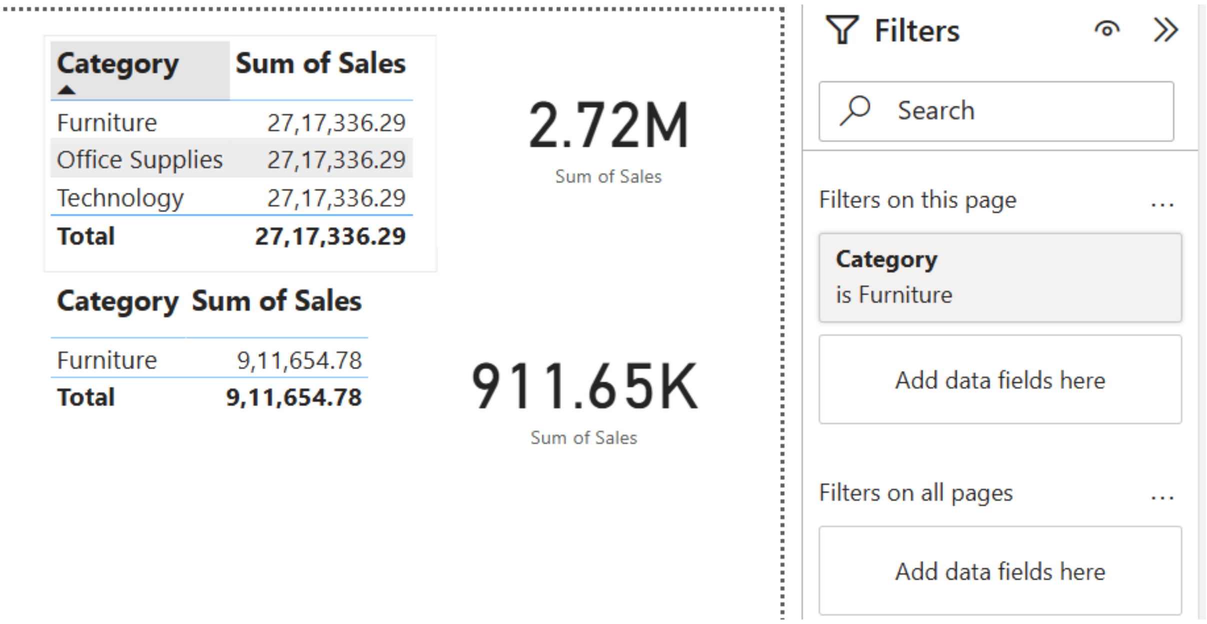


```
let
    Source = Sql.Databases("."),
    Practice = Source{[Name="Practice"]}[Data],
    dbo_Orders = Practice{[Schema="dbo",Item="Orders"]}[Data]
in
    dbo_Orders
```

✓ No syntax errors have been detected.

Dashboard & Reporting

Remove Page Level Filtration on Specific Visuals Using Quisk Measure



Category	Sum of Sales
Furniture	27,17,336.29
Office Supplies	27,17,336.29
Technology	27,17,336.29
Total	**27,17,336.29**

2.72M

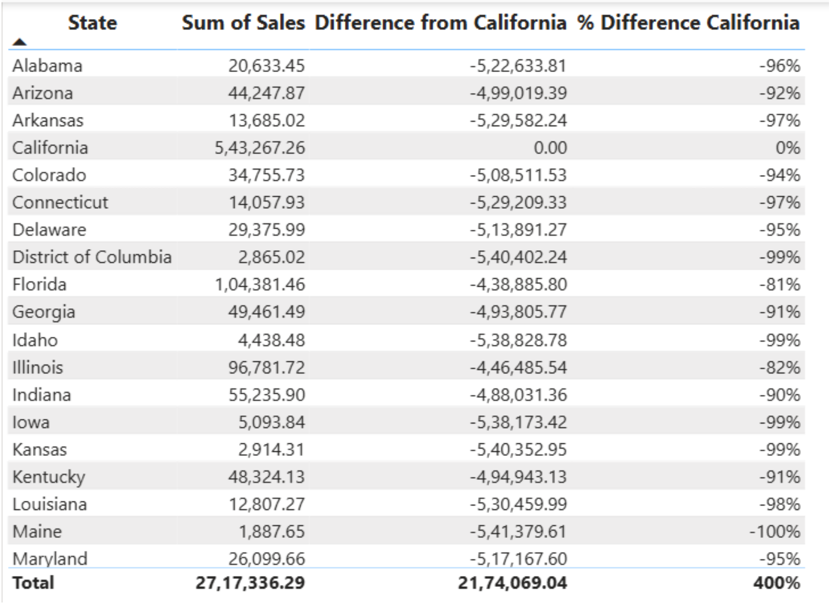
Sum of Sales

Category	Sum of Sales
Furniture	9,11,654.78
Total	**9,11,654.78**

911.65K

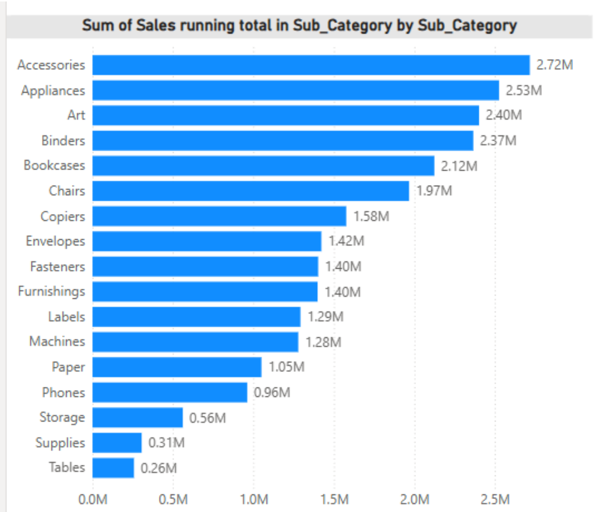
Sum of Sales

Quick Measures - Comparison:

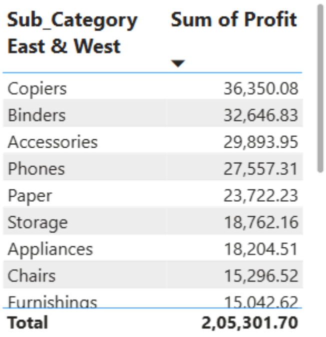


State	Sum of Sales	Difference from California	% Difference California
Alabama	20,633.45	-5,22,633.81	-96%
Arizona	44,247.87	-4,99,019.39	-92%
Arkansas	13,685.02	-5,29,582.24	-97%
California	5,43,267.26	0.00	0%
Colorado	34,755.73	-5,08,511.53	-94%
Connecticut	14,057.93	-5,29,209.33	-97%
Delaware	29,375.99	-5,13,891.27	-95%
District of Columbia	2,865.02	-5,40,402.24	-99%
Florida	1,04,381.46	-4,38,885.80	-81%
Georgia	49,461.49	-4,93,805.77	-91%
Idaho	4,438.48	-5,38,828.78	-99%
Illinois	96,781.72	-4,46,485.54	-82%
Indiana	55,235.90	-4,88,031.36	-90%
Iowa	5,093.84	-5,38,173.42	-99%
Kansas	2,914.31	-5,40,352.95	-99%
Kentucky	48,324.13	-4,94,943.13	-91%
Louisiana	12,807.27	-5,30,459.99	-98%
Maine	1,887.65	-5,41,379.61	-100%
Maryland	26,099.66	-5,17,167.60	-95%
Total	**27,17,336.29**	**21,74,069.04**	**400%**

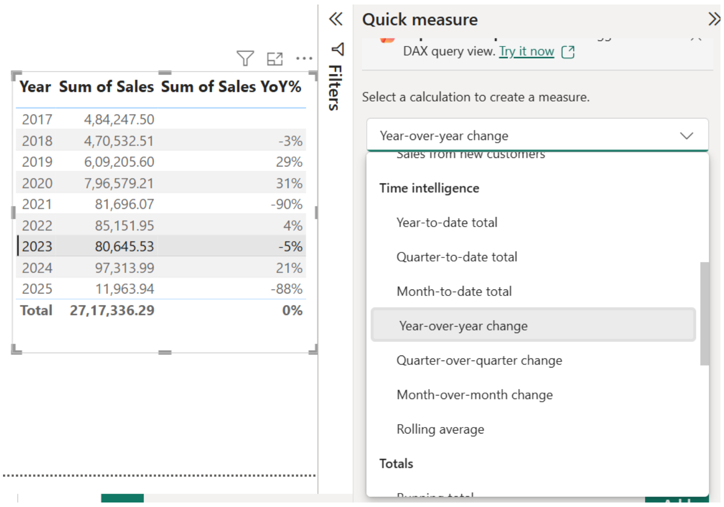
Dashboard & Reporting



2.72M
Total_Sale_Without_Discount



Sub_Category East & West	Sum of Profit
Copiers	36,350.08
Binders	32,646.83
Accessories	29,893.95
Phones	27,557.31
Paper	23,722.23
Storage	18,762.16
Appliances	18,204.51
Chairs	15,296.52
Furnishings	15,042.62
Total	**2,05,301.70**



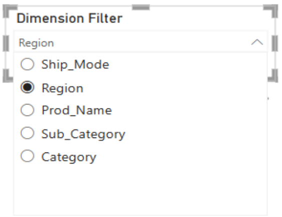
Year	Sum of Sales	Sum of Sales YoY%
2017	4,84,247.50	
2018	4,70,532.51	-3%
2019	6,09,205.60	29%
2020	7,96,579.21	31%
2021	81,696.07	-90%
2022	85,151.95	4%
2023	80,645.53	-5%
2024	97,313.99	21%
2025	11,963.94	-88%
Total	**27,17,336.29**	**0%**

DAX query view. Try it now

Select a calculation to create a measure.

Year-over-year change

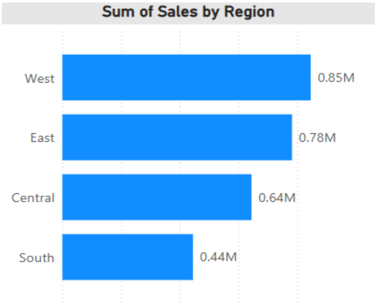
Dimension Filtering Using New Parameter Fields

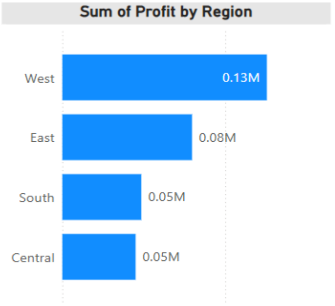


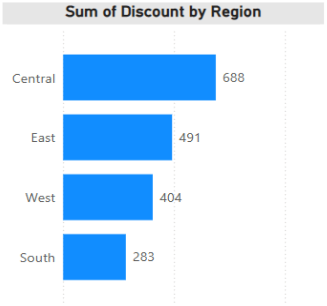
Dimension Filter

Region

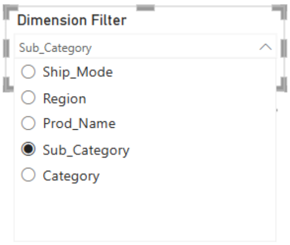
○ Ship_Mode
● Region
○ Prod_Name
○ Sub_Category
○ Category







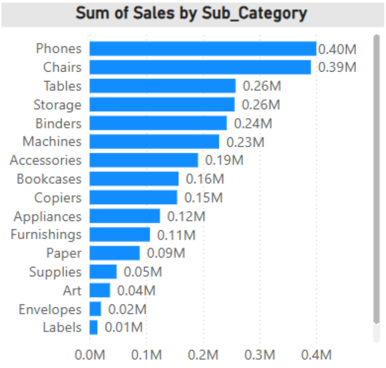
Dashboard & Reporting

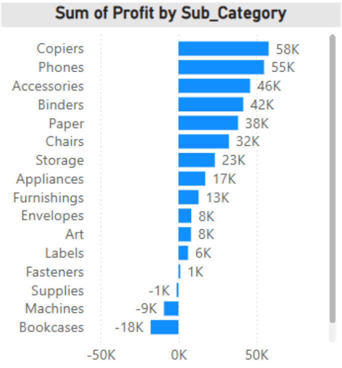


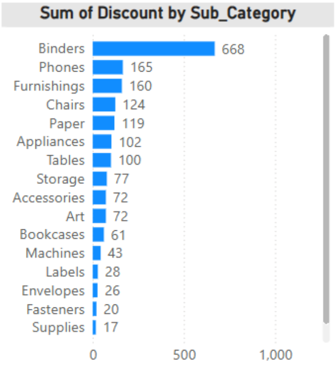
Dimension Filter

Sub_Category

○ Ship_Mode
○ Region
○ Prod_Name
● Sub_Category
○ Category

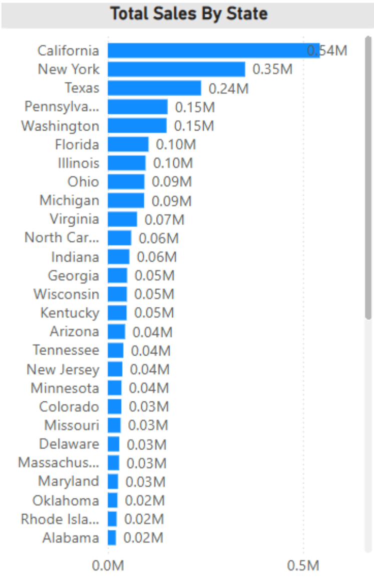


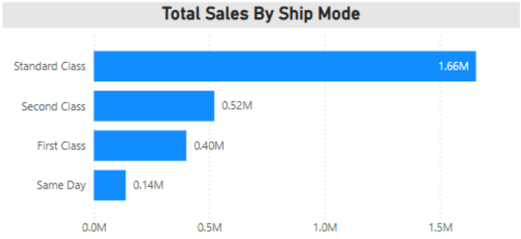


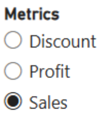


Measure Field Filtering

Sales:

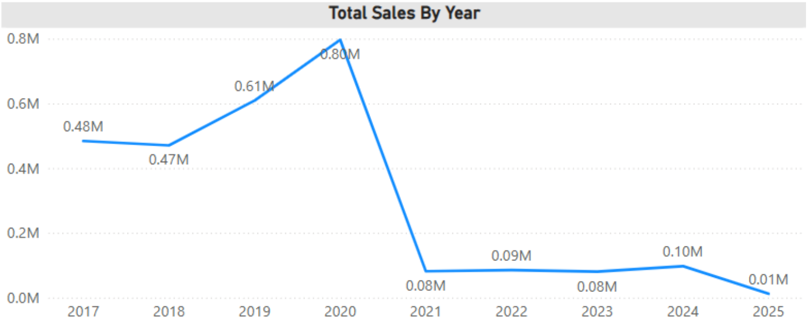






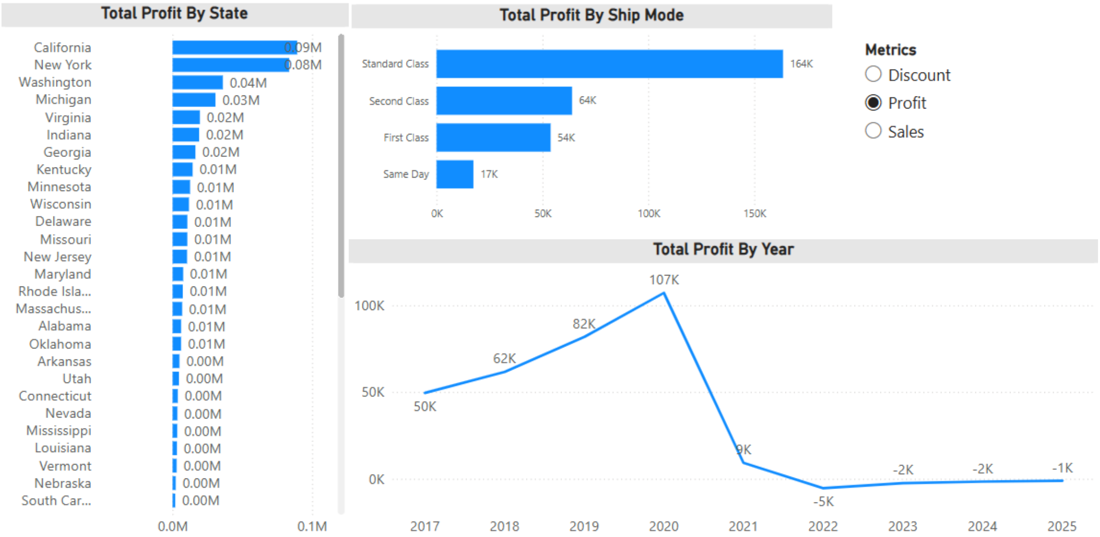
Metrics

○ Discount
○ Profit
● Sales



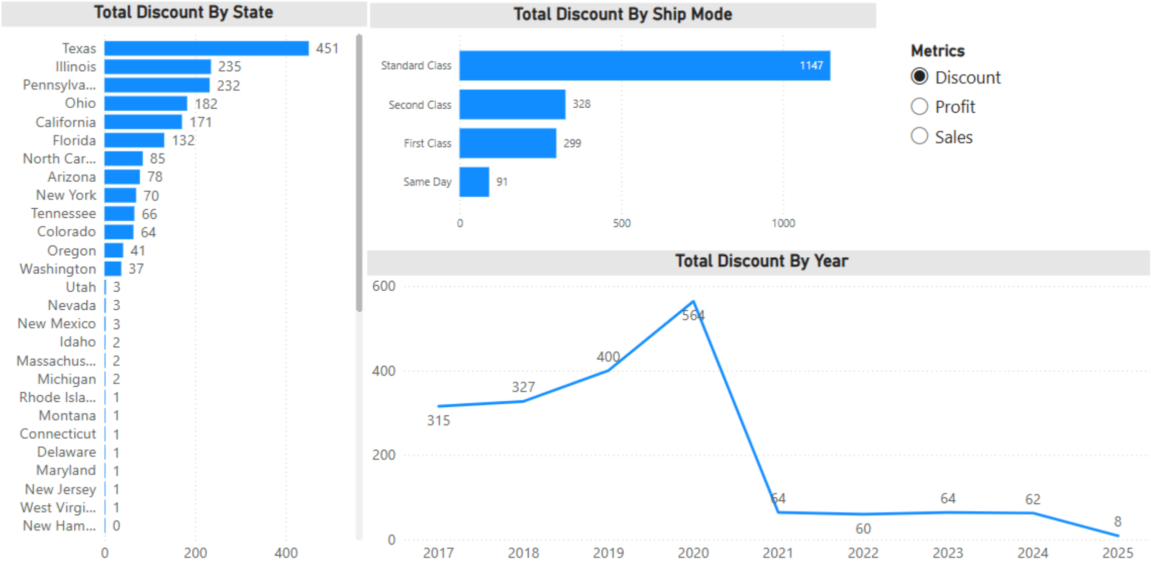
Dashboard & Reporting

Profit:



Metrics
○ Discount
● Profit
○ Sales

Discount:



Metrics
● Discount
○ Profit
○ Sales

Create a new Table: With Measure Fileds
Column1 Metrics - Sales, Profit, Discount

Column2 Measures_Selection =

```
        switch(True(),

        DISTINCT(Measure_Selection[Measure_Fields]) = "Sales", sum(OrdersSQL[Sales]),
        DISTINCT(Measure_Selection[Measure_Fields]) = "Profit", sum(OrdersSQL[Profit]),
        DISTINCT(Measure_Selection[Measure_Fields]) = "Discount", sum(OrdersSQL[Discount]))
```

To update the Title based on selection use below format title option:

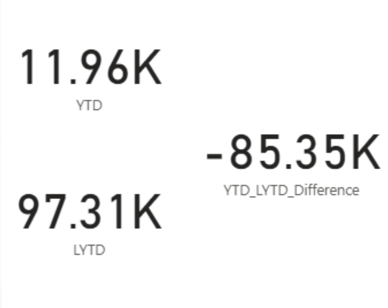
Selected_Value = "Total " & SELECTEDVALUE(Measure_Selection[Metrics]) & "By State"

Dashboard & Reporting

DAX: Use Of Variable to Optimise Performance

```
YTD_LYTD_Difference =

        VAR TYTD = TOTALYTD(SUM(Orders[Sales]), Orders[OrderDate])

        VAR LYTD = TOTALYTD(SUM(Orders[Sales]),
        PARALLELPERIOD(Orders[OrderDate].[Date], -1, Year))

        return

        TYTD-LYTD
```

```
YTD_Calc = TOTALYTD(SUM(Orders[Sales]), Orders[OrderDate])

LYTD_Calc = TOTALYTD(SUM(Orders[Sales]), PARALLELPERIOD(Orders[OrderDate].[Date],
-1, Year))
```



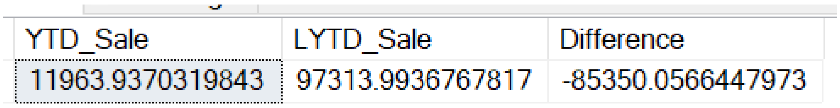
11.96K
YTD

-85.35K
YTD_LYTD_Difference

97.31K
LYTD

Validate:

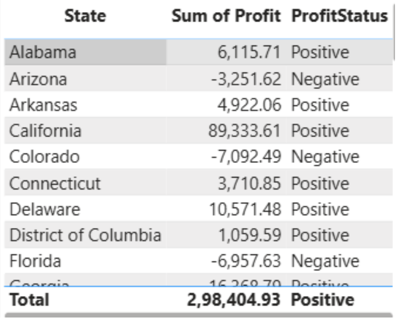
```
with cte as (select SUM(Sales) as YTD_Sale
from Orders where Datediff(Year, OrderDate, GETDATE()) = 0),
cte2 as (select SUM(Sales) as LYTD_Sale
from Orders where Datediff(Year, OrderDate, GETDATE()) = 1)
select a.YTD_Sale, b.LYTD_Sale, a.YTD_Sale-b.LYTD_Sale as Difference from cte a
cross join cte2 b;
```



YTD_Sale	LYTD_Sale	Difference
11963.9370319843	97313.9936767817	-85350.0566447973

IF:

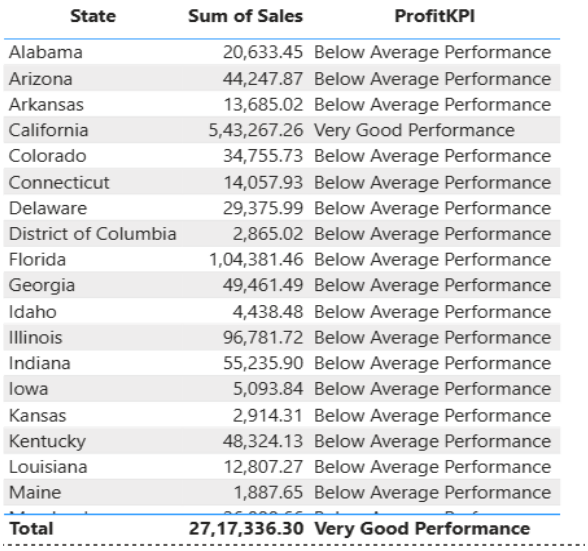
```
ProfitStatus = IF(SUM(Orders[Profit]) > 0, "Positive", "Negative")
```



State	Sum of Profit	ProfitStatus
Alabama	6,115.71	Positive
Arizona	-3,251.62	Negative
Arkansas	4,922.06	Positive
California	89,333.61	Positive
Colorado	-7,092.49	Negative
Connecticut	3,710.85	Positive
Delaware	10,571.48	Positive
District of Columbia	1,059.59	Positive
Florida	-6,957.63	Negative
Georgia	16,268.70	Positive
Total	**2,98,404.93**	**Positive**

Dashboard & Reporting

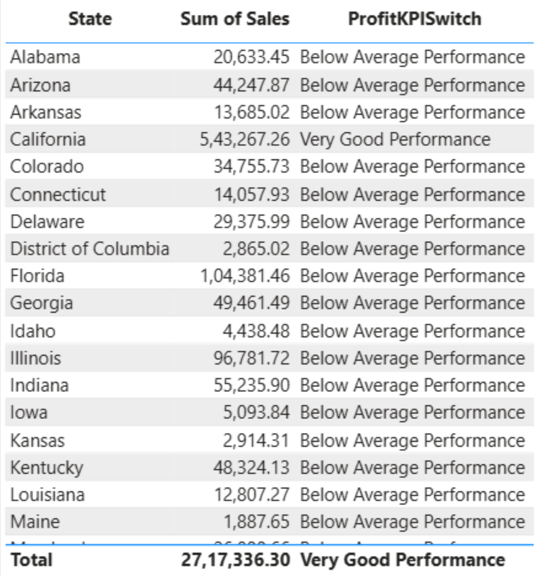
```
ProfitKPI = IF(SUM(Orders[Sales]) > 300000, "Very Good Performance",

        IF(SUM(Orders[Sales]) > 250000, "Good Performance",
           IF(SUM(Orders[Sales]) > 190000, "Above Average Performance",
               IF(SUM(Orders[Sales])> 160000, "Average Performance",
                  IF(SUM(Orders[Sales])> 125000, "Average", "Bad Performance")
                              ))))
```



State	Sum of Sales	ProfitKPI
Alabama	20,633.45	Below Average Performance
Arizona	44,247.87	Below Average Performance
Arkansas	13,685.02	Below Average Performance
California	5,43,267.26	Very Good Performance
Colorado	34,755.73	Below Average Performance
Connecticut	14,057.93	Below Average Performance
Delaware	29,375.99	Below Average Performance
District of Columbia	2,865.02	Below Average Performance
Florida	1,04,381.46	Below Average Performance
Georgia	49,461.49	Below Average Performance
Idaho	4,438.48	Below Average Performance
Illinois	96,781.72	Below Average Performance
Indiana	55,235.90	Below Average Performance
Iowa	5,093.84	Below Average Performance
Kansas	2,914.31	Below Average Performance
Kentucky	48,324.13	Below Average Performance
Louisiana	12,807.27	Below Average Performance
Maine	1,887.65	Below Average Performance
Total	**27,17,336.30**	**Very Good Performance**

We can achieve the same using SWITCH

```
ProfitKPISwitch = SWITCH(True(),

            SUM(Orders[Sales]) > 300000, "Very Good Performance",
            SUM(Orders[Sales]) > 250000, "Good Performance",
            SUM(Orders[Sales]) > 190000, "Above Average Performance",
            SUM(Orders[Sales])> 160000, "Average Performance",
            SUM(Orders[Sales]) > 125000, "Average Performance",
                "Bad Performance")
```

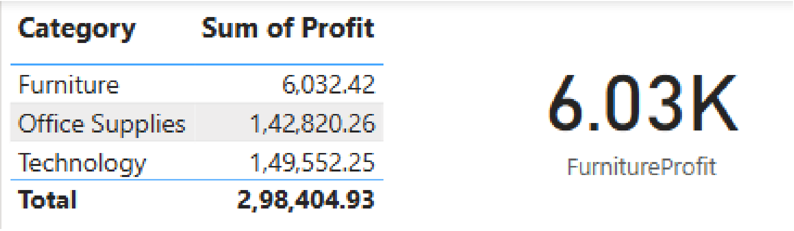


State	Sum of Sales	ProfitKPISwitch
Alabama	20,633.45	Below Average Performance
Arizona	44,247.87	Below Average Performance
Arkansas	13,685.02	Below Average Performance
California	5,43,267.26	Very Good Performance
Colorado	34,755.73	Below Average Performance
Connecticut	14,057.93	Below Average Performance
Delaware	29,375.99	Below Average Performance
District of Columbia	2,865.02	Below Average Performance
Florida	1,04,381.46	Below Average Performance
Georgia	49,461.49	Below Average Performance
Idaho	4,438.48	Below Average Performance
Illinois	96,781.72	Below Average Performance
Indiana	55,235.90	Below Average Performance
Iowa	5,093.84	Below Average Performance
Kansas	2,914.31	Below Average Performance
Kentucky	48,324.13	Below Average Performance
Louisiana	12,807.27	Below Average Performance
Maine	1,887.65	Below Average Performance
Total	**27,17,336.30**	**Very Good Performance**

Dashboard & Reporting

Calculate: If you want to filter and control what Expression has to count or aggregate then we can use Calculate which allows you to use more than one condition within a expression

```
FurnitureProfit = CALCULATE(SUM(Orders[Profit]), Orders[Category] = "Furniture")
```



Category	Sum of Profit
Furniture	6,032.42
Office Supplies	1,42,820.26
Technology	1,49,552.25
Total	**2,98,404.93**

6.03K

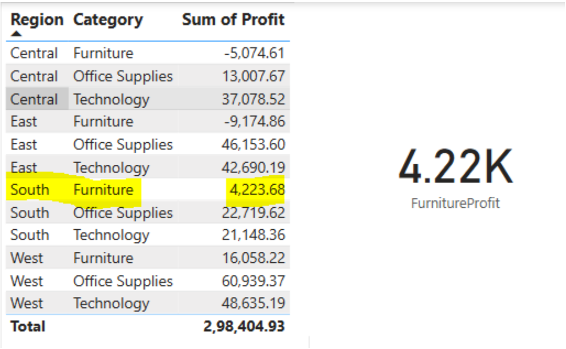
FurnitureProfit

Why CALCULATE Is Important

- Allows dynamic filtering within measures.
- Works well for **time intelligence** calculations.
- Can combine multiple criteria efficiently.

Further_Filtration

```
FurnitureProfit = CALCULATE(SUM(Orders[Profit]), Orders[Category] = "Furniture" &&
Orders[Region] = "South")
```



Region	Category	Sum of Profit
Central	Furniture	-5,074.61
Central	Office Supplies	13,007.67
Central	Technology	37,078.52
East	Furniture	-9,174.86
East	Office Supplies	46,153.60
East	Technology	42,690.19
South	Furniture	4,223.68
South	Office Supplies	22,719.62
South	Technology	21,148.36
West	Furniture	16,058.22
West	Office Supplies	60,939.37
West	Technology	48,635.19
Total		**2,98,404.93**

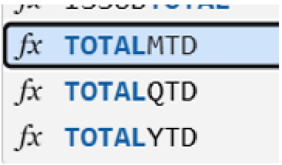
4.22K

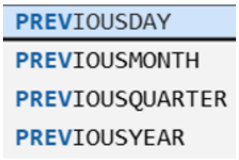
FurnitureProfit

Time Intelligence Functions:

Combine below functions

PARALLELPERIOD(Orders[OrderDate].[Date], -1, year) LYTD





Dashboard & Reporting

Compare Year Over Year Change:

Previous Year – Last 1 Year

```
YOY Comparsion = SUM(Orders[Sales])-CALCULATE(Sum(Orders[Sales]),
PREVIOUSYEAR(Orders[OrderDate].[Date]))
```

ParallelPeriod – previous to last 2 or more

```
YOY Comparsion = SUM(Orders[Sales])-CALCULATE(Sum(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -1, year))
```

Compare Same Month Last year

```
Same Month Last Year = SUM(Orders[Sales])-CALCULATE(SUM(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -12,MONTH))
```

QOQ Change:

```
QOQ Change = SUM(Orders[Sales])-CALCULATE(SUM(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -1, QUARTER))
```

MOM Change:

```
MOM Change = SUM(Orders[Sales])-CALCULATE(SUM(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -1, MONTH))
```

YTD_Sale:

```
YTD_Sale = TOTALYTD(SUM(Orders[Sales]), Orders[OrderDate])
```

LYTD_Sale:

```
LYTD_Sale = TOTALYTD(SUM(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -1, Year))
```

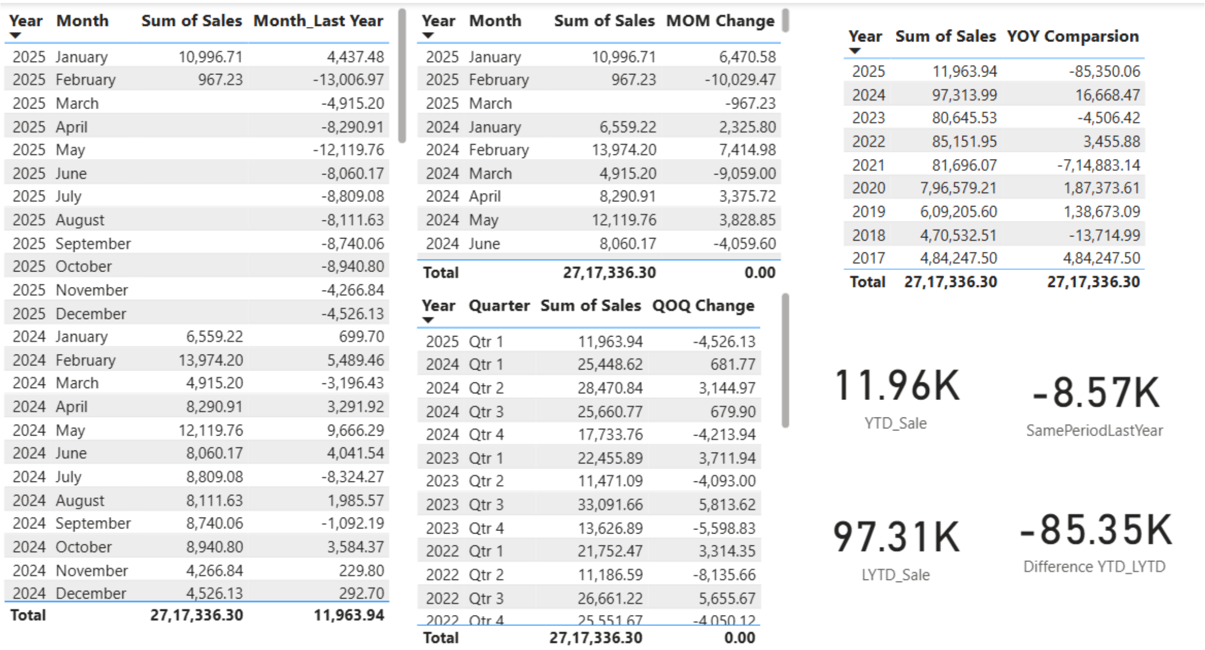
Difference YTD & LYT:

```
Difference YTD_LYTD =

var YTD_Sales = TOTALYTD(SUM(Orders[Sales]), Orders[OrderDate])

var LYTD_Sales = TOTALYTD(SUM(Orders[Sales]),
PARALLELPERIOD(Orders[OrderDate].[Date], -1, year))

Return
YTD_Sales - LYTD_Sales
```

Dashboard & Reporting

Same_Period_LastYear:

```
SamePeriodLastYear = TOTALYTD(SUM(Orders[Sales]), Orders[OrderDate])-
CALCULATE(SUM(Orders[Sales]),
DATESYTD(SAMEPERIODLASTYEAR(Orders[OrderDate])))
```



Year	Month	Sum of Sales	Month_Last Year
2025	January	10,996.71	4,437.48
2025	February	967.23	-13,006.97
2025	March		-4,915.20
2025	April		-8,290.91
2025	May		-12,119.76
2025	June		-8,060.17
2025	July		-8,809.08
2025	August		-8,111.63
2025	September		-8,740.06
2025	October		-8,940.80
2025	November		-4,266.84
2025	December		-4,526.13
2024	January	6,559.22	699.70
2024	February	13,974.20	5,489.46
2024	March	4,915.20	-3,196.43
2024	April	8,290.91	3,291.92
2024	May	12,119.76	9,666.29
2024	June	8,060.17	4,041.54
2024	July	8,809.08	-8,324.27
2024	August	8,111.63	1,985.57
2024	September	8,740.06	-1,092.19
2024	October	8,940.80	3,584.37
2024	November	4,266.84	229.80
2024	December	4,526.13	292.70
Total		27,17,336.30	11,963.94

Year	Month	Sum of Sales	MOM Change
2025	January	10,996.71	6,470.58
2025	February	967.23	-10,029.47
2025	March		-967.23
2024	January	6,559.22	2,325.80
2024	February	13,974.20	7,414.98
2024	March	4,915.20	-9,059.00
2024	April	8,290.91	3,375.72
2024	May	12,119.76	3,828.85
2024	June	8,060.17	-4,059.60
Total		27,17,336.30	0.00

Year	Quarter	Sum of Sales	QOQ Change
2025	Qtr 1	11,963.94	-4,526.13
2024	Qtr 1	25,448.62	681.77
2024	Qtr 2	28,470.84	3,144.97
2024	Qtr 3	25,660.77	679.90
2024	Qtr 4	17,733.76	-4,213.94
2023	Qtr 1	22,455.89	3,711.94
2023	Qtr 2	11,471.09	-4,093.00
2023	Qtr 3	33,091.66	5,813.62
2023	Qtr 4	13,626.89	-5,598.83
2022	Qtr 1	21,752.47	3,314.35
2022	Qtr 2	11,186.59	-8,135.66
2022	Qtr 3	26,661.22	5,655.67
2022	Qtr 4	25,551.67	-4,050.12
Total		27,17,336.30	0.00

Year	Sum of Sales	YOY Comparsion
2025	11,963.94	-85,350.06
2024	97,313.99	16,668.47
2023	80,645.53	-4,506.42
2022	85,151.95	3,455.88
2021	81,696.07	-7,14,883.14
2020	7,96,579.21	1,87,373.61
2019	6,09,205.60	1,38,673.09
2018	4,70,532.51	-13,714.99
2017	4,84,247.50	4,84,247.50
Total	27,17,336.30	27,17,336.30

11.96K
YTD_Sale

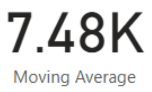
-8.57K
SamePeriodLastYear

97.31K
LYTD_Sale

-85.35K
Difference YTD_LYTD

Moving Average:

```
Moving Average =

var Latest_Date = MAX(Orders[OrderDate])

var Start_Date = EDATE(Latest_Date, -6)

var Sales6Month = CALCULATE(SUM(Orders[Sales]),
FILTER(ALL(Orders[OrderDate]), Orders[OrderDate] > Start_Date &&
Orders[OrderDate] < Latest_Date))

VAR Number_Months = 6

RETURN

ROUND(DIVIDE(Sales6Month, Number_Months),2)
```



7.48K
Moving Average

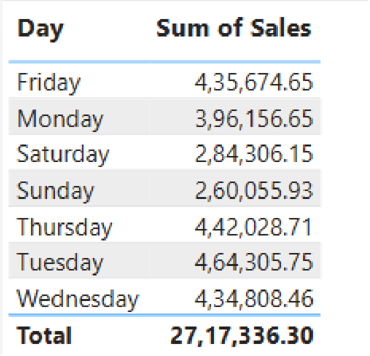
Dashboard & Reporting

Create a calculated column with WeekDay and also Weekday Number:

```
Daynum = WEEKDAY(Orders[OrderDate])
```

Note: Summarisation: Don’t Summarise

```
Day = FORMAT(Orders[OrderDate], "dddd") Extract Week_Day
```

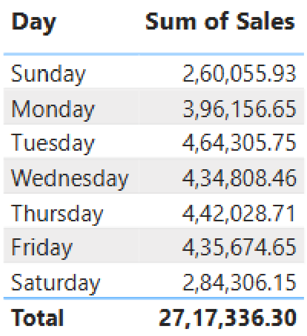


Day	Sum of Sales
Friday	4,35,674.65
Monday	3,96,156.65
Saturday	2,84,306.15
Sunday	2,60,055.93
Thursday	4,42,028.71
Tuesday	4,64,305.75
Wednesday	4,34,808.46
Total	**27,17,336.30**

In order to Sort the values as per Weekday:

Select Day Column from the Data:

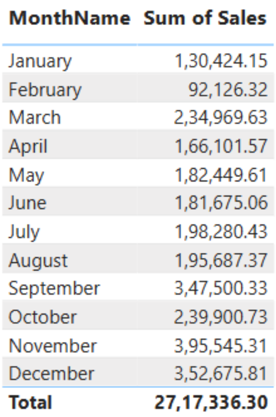
Sort By Column > Select DayNum Column that we have created



Day	Sum of Sales
Sunday	2,60,055.93
Monday	3,96,156.65
Tuesday	4,64,305.75
Wednesday	4,34,808.46
Thursday	4,42,028.71
Friday	4,35,674.65
Saturday	2,84,306.15
Total	**27,17,336.30**

Month:

```
MonthName = FORMAT(Orders[OrderDate], "MMMM")

MonNum = MONTH(Orders[OrderDate])
```

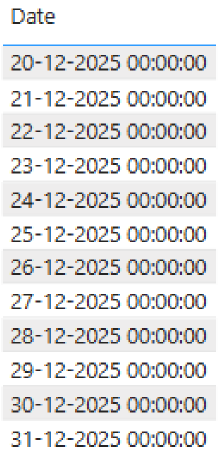


MonthName	Sum of Sales
January	1,30,424.15
February	92,126.32
March	2,34,969.63
April	1,66,101.57
May	1,82,449.61
June	1,81,675.06
July	1,98,280.43
August	1,95,687.37
September	3,47,500.33
October	2,39,900.73
November	3,95,545.31
December	3,52,675.81
Total	**27,17,336.30**

Dashboard & Reporting

Generate dates:

```
Date_Calendar = CALENDAR("2025-01-01", "2025-12-31")
```

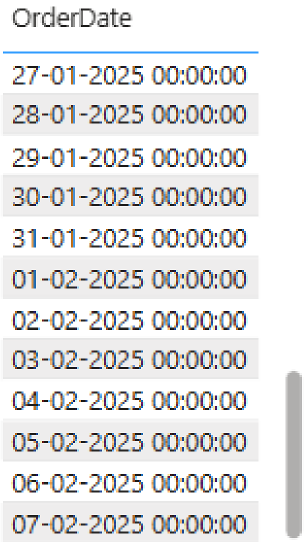


Date
20-12-2025 00:00:00
21-12-2025 00:00:00
22-12-2025 00:00:00
23-12-2025 00:00:00
24-12-2025 00:00:00
25-12-2025 00:00:00
26-12-2025 00:00:00
27-12-2025 00:00:00
28-12-2025 00:00:00
29-12-2025 00:00:00
30-12-2025 00:00:00
31-12-2025 00:00:00

Datesbetween:

Need Date field from our data as reference, this will create dates as per our data which are within the given range

```
DATESBETWEEN(Orders[OrderDate], "2025-01-01", "2025-12-31")
```



OrderDate
27-01-2025 00:00:00
28-01-2025 00:00:00
29-01-2025 00:00:00
30-01-2025 00:00:00
31-01-2025 00:00:00
01-02-2025 00:00:00
02-02-2025 00:00:00
03-02-2025 00:00:00
04-02-2025 00:00:00
05-02-2025 00:00:00
06-02-2025 00:00:00
07-02-2025 00:00:00

Dashboard & Reporting

Moving Average Sales =

```
var Latest_Date = MAX(Orders[OrderDate])

var Start_Date = EDATE(Latest_Date, -6)

var Sales6Month = CALCULATE(SUM(Orders[Sales]), DATESINPERIOD(Orders[OrderDate],
Latest_Date, -6, MONTH))

VAR Number_Months = 6

RETURN

ROUND(DIVIDE(Sales6Month, Number_Months),2)
```

Moving Average Profit:

```
Rolling_Average_Profit_6Months =


var Latest_Date = LASTDATE(Orders[OrderDate])

var sixmonthProfit = CALCULATE(sum(Orders[Profit]),
DATESINPERIOD(Orders[OrderDate], Latest_Date, -6, MONTH))

return

ROUND(DIVIDE(sixmonthProfit, 6),2)
```

ALL():

which help fetch all the values from the selected columns, helpful to avoid page level or report level filtration within the visuals:

```
Total_Sale = CALCULATE(sum(Orders[Sales]), ALL(Orders[Region], Orders[OrderDate]))
```

ALLExcept(): *To Optimize performance when we are filtering many columns*

Considers filtering on selected columns and ignores filtering on any filed that is not selected

```
Total_Sale = CALCULATE(sum(Orders[Sales]), ALLEXCEPT(Orders, Orders[Ship Mode]))
```

Dashboard & Reporting

Ensure we have relation created and utilize below functions to enhance performance

Related(): Cardinality - Many to 1

```
Related(Prod[Cost])

Using related we can utilize columns of a different table which has relation ship
Example:

ONE TO MANY Relationship
Dimnessions to Fact Table
```

RelatedTable(): One to Many Relation Fact to Dimensions

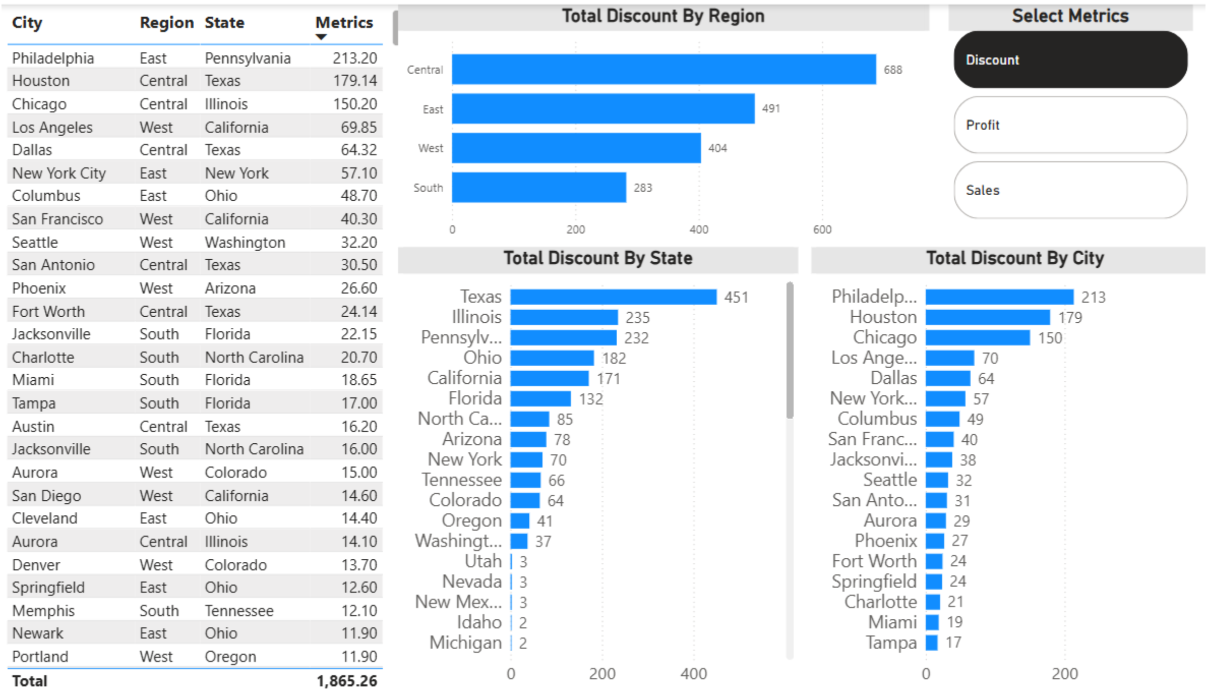
```
Input parameter as measure Field

(TableName, Measurefield)
SUMX(prod_Sales, prod_Sales[Price])

SUMX COUNTX MAXX MINX AVGX (Row-Level operations)


RelatedTable(Prod_Sales, prod_Sales[Price])
```

CrossFilter(Table1, Table2, Direction):

```
TotalSales_ForceFilter =
CALCULATE(SUM(Orders[Sales]), CROSSFILTER(Orders[CustomerID],
Customers[CustomerID], BOTH))
```

Dashboard & Reporting

Dashborad with Selection control on Measure Fields : Sales, Profit, Discount



City	Region	State	Metrics
Philadelphia	East	Pennsylvania	213.20
Houston	Central	Texas	179.14
Chicago	Central	Illinois	150.20
Los Angeles	West	California	69.85
Dallas	Central	Texas	64.32
New York City	East	New York	57.10
Columbus	East	Ohio	48.70
San Francisco	West	California	40.30
Seattle	West	Washington	32.20
San Antonio	Central	Texas	30.50
Phoenix	West	Arizona	26.60
Fort Worth	Central	Texas	24.14
Jacksonville	South	Florida	22.15
Charlotte	South	North Carolina	20.70
Miami	South	Florida	18.65
Tampa	South	Florida	17.00
Austin	Central	Texas	16.20
Jacksonville	South	North Carolina	16.00
Aurora	West	Colorado	15.00
San Diego	West	California	14.60
Cleveland	East	Ohio	14.40
Aurora	Central	Illinois	14.10
Denver	West	Colorado	13.70
Springfield	East	Ohio	12.60
Memphis	South	Tennessee	12.10
Newark	East	Ohio	11.90
Portland	West	Oregon	11.90
Total			**1,865.26**

Future Scope: Custom Tooltip with visuals, Bookmarks