JARGONS

data visualization tools

PowerBI

Requirement must have a dimension, time attribute, sometimes display tooltip.

Data-types:

Number-Integer ,Real, Money, Currency

Character/String: Alphabets, Words

Date: dd/mm/yyyy ;dd-mon-yyyy; mm/dd/yyyy ;

Logical/Boolean: 0 and 1, True or False , Yes or No etc.

Quantitative data type: can do mathematical operations, Mathematical operations are possible on dates.

Qualitative data type: char/ string, numerical(no meaning to mathematical operation.)

Concepts related to tables (RDBMS- Oracle,SQL,MySQL):

Primary Key: All value must be distinct, There should not be blank value assigned to the p.k coloumn(NOT NULL).

Unique Key : Uniquely identified coloumns(MAY HAVE NULL VALUES)

Candidate/Alternate Key: Can keep it as primary key of the main p.k data is deleted.

General Form:

Clustered Coloumn Chart:

Y-Axis:Measures (Value Field)

X-Axis:Dimension

Clustered Bar Chart:

X and Y axis get reversed with reference to clustered coloumn chart.

Y-Axis:Dimension

X-Axis:Measures (Value Field)

Line Chart:

Generally time related data is always line chart.

FOR FILTERINGS (by default)

TEXT TYPE - BASIC FILTERING

NUMBER TYPE - ADVANCE FILTERING

When value field is going to change it is basic filterings (user selected)

When value field is going to remain constant it is advance filterings.

When there are multiple dimesnsions given then use stacked coloumn chart or stacked bar chart.

For time related questions generally use line chart.

Whenever we have to use count filter always use count(distinct).

Whenever percentage is given generally use pie chart.

WHEN TO USE MATRIX CHART OR TABLE CHART.

1.)TABLE:-Use a table when you want to show flat data, such as a list of customers and their contact details. Tables are also useful when you want to save space on your report or dashboard.

2.)MATRIX:-Use a matrix when you want to show hierarchical data with multiple levels of detail, such as sales by region, country, city, and product. Matrices are also useful when you want to show subtotals and grand totals for your data.

HIERARCHY:

Hierarchies allow users to explore relationships between data points and drill down into data.

Maintain consistent hierarchical structure

You can incorporate hierarchies into your semantic model so that all reports built from that model maintain the same hierarchical structure.

DAX Functions:(Data Analysis Expressions)

They are used to create a new column along with different formulae.

They Use special Symbol For DAX (Created Column) by which we are able to know that this is created column.

Similarly for HIERARCHY.

KPI: Key Performance Indicator.

Measure:Summarized Calculations

DAX FUNCTIONS:

To concatinate two columns:

col\_name= 'main data'[heirarchy\_name].[Col\_Name] & "-" & 'main data' [hierarchy\_name].[coloumn\_name].

For Example:

Year Month Name = 'Main Data'[Order Date].[Month] & "-" & 'Main Data'[Order Date].[Year]

LEFT is used to select first 3 letters of the data..

For Example:

Year Month Name = LEFT('Main Data'[Order Date].[Month],3) & "-" & 'Main Data'[Order Date].[Year]

If the condition is satisfying without sorting then let it be like that only...but if sorting is not possible then create another column and sort by that column.

DYNAMIC TITLE: Dynamic titles provide immediate context to users, informing them about the specific subset of data they are analyzing

Create a Measure

Then

For Example... (for single selection)

Chart Title = "Sales by Region For Year : " & VALUES('Main Data'[Order Date].[Year]) & ", For Market : " & VALUES('Main Data'[Market])

Then put this in title f(x) in that 2nd row and 2nd column.

To create a dynamic chart with multiple selections the synatx with example is

Dynamic Table Chart = "03-Sales by Region for Year: " & VALUES('Main Data'[Order Date].[Year]) & ", For Market : " & CONCATENATEX(VALUES('Main Data'[Market]), 'Main Data'[Market], "," , 'Main Data'[Market],ASC)

When we have to calculate row wise per record then new coloumn is created

and when we have to find total the we use new measure.

Ellapsed Time:- The time which has gone. Eg: time difference between order date and delivery date

Table Relationships:

Same as SQL.

If you query two or more tables at the same time, when the data is loaded, Power BI Desktop attempts to find and create relationships for you.

One to many :- 1< many:-\*

When we have to connect the columns from to different dataset in which there is no relation the we have to create a common column in both the dataset.

Always take table from one to many madhun one madhun karaycha for visualization.

REMOVEFILTERS can only be used to clear filters but not to return a table.

REMOVEFILTERS madhe write the coloumn name which is summarization dimension OR External Filters. Yha doghan madhlach asnare.

CALCULATE Function is used to calculate the created measure.The CALCULATE function is basically just an abstraction of the FILTER function in combination with the ALL function.

ALLEXCEPT - Removes all context filters in the table except filters that have been applied to the specified columns.It means remove all filters except the coloumn you need.

The values of both the above two will give the same result.

DIMESNSION WILL ALWAYS BE A ROW CONTEXT

EXTERNAL FILTER WILL ALWAYS BE A FILTER CONTEXT

In DAX, row context and filter context are two different ways to apply context to a formula, and they differ in how they operate, what they apply to, and how they are created:

Row context

Operates on a single row at a time, and is determined by the table's structure. It applies to the current row of a table or column, and includes the values in that row and related columns. Row context is used when operating on values in a single row of a table.

Filter context

Operates on a set of rows at once, and is determined by the user's interaction with the data. It applies to the entire data model based on active filters. Filter context changes how data is aggregated and/or scanned.

IF Functions

same as the condition in c programming.

Related is written because we have tell that it is related table coloumn.

ALLCROSSFILTERS: Function in which it removes all the filters.

ALLCROSSFILTERED removes all the filters on an expanded table (like ALL) and on columns and tables that are cross-filtering the table argument because of limited (weak) relationships and/or bidirectional cross-filters set on relationships directly or indirectly connected to the expanded table.

Contribution sathi kasa karaycha:- individual sales done by person / total sales \*100.

yearly sales = Yearly Sale = CALCULATE (SUM('Main Data'[Sales]),ALLEXCEPT('Main Data’, ‘Main Data'[Order Date].[Year]))

Whenever "average" is mentioned in the requirement always use distinct count in denominator.

Whenever "per" is used in requirement then use remove filter’s function is used.

Target Deviation Percentage = (SUM ('Main Data'[Sales])-SUM ('Target Data'[Target]))/SUM ('Target Data'[Target])

General Formula for target and sales related data.

Dimensions are required when there is a parameter for e.g. wise etc.

When slicer visualization is used, filters when applied such as order date or market etc will get applied to the same page.

IF-ELSE in DAX:

Colour Code For Sales = IF(SUM('Main Data'[Sales])>=1.25 \* [Average Sale per State],"Blue”, IF (SUM('Main Data'[Sales])<=0.9 \* [Average Sale per State],"Dark Blue”, “Sky Blue"))

IF (CONDITION, RESULTIFTRUE, RESULTIFFALSE)

COLOURS USING DIFFERENT FIELD MADHE: YOU HAVE TO SELECT THE COLUMN THEN GO IN F(x) and change to gradient.

TO CREATE A NEW TABLE USING AGGREGATE FUNCTION

General Form:

SUMMARIZE (**Table**, [GroupBy\_ColumnName1], ..., [Name1], [Expression1], ...)

SUMMARIZE(Table, [GroupBy\_ColumnName1], ..., [GroupBy\_ColumnName3], [Name1], **[Expression1]**, ...)

Creates a summary the input table grouped by the specified columns.

EXAMPLE:

20 01 = SUMMARIZE ('Main Data’, ‘Main Data'[Order Date]. [Year],'Main Data'[Market],'Main Data'[Region],"Year Market Region Profit”, SUM ('Main Data'[Profit]))

SUMX(**Table**, Expression)

Returns the sum of an expression evaluated for each row in a table.

It is SUM with Some CONDITION.

FILTER(**Table**, FilterExpression)

Returns a table that has been filtered

EARLIER(**ColumnName**, [Number])

Returns the value in the column prior to the specified number of table scans (default is 1). Earlier mhanje kuthlhi measuring value chi sum karat raha, karat raha until and unless to year change hot nahi toparaynt. Previous Record and

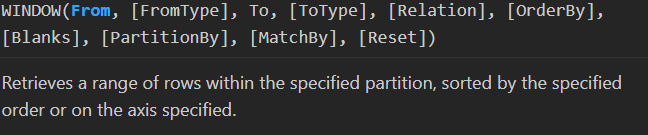
Profit For Market = SUMX(FILTER('New Data','New Data'[Year] = EARLIER('New Data'[Year]) && 'New Data'[Market] = EARLIER('New Data'[Market])),'New Data'[Year Market Region Profit])

Month over Month sales is the difference between sales of two months

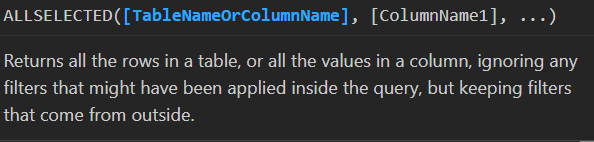
In-built function ahe previous month mhanun.

Previous Month function works only on dates.

Cumulative Sales:



ALLSELECTED Function:



REQUIREMENT NUMBER 20-1-02 PowerBI Explanation

1. **SUMX**:
   * **Purpose**: SUMX is an iterator function. It loops through each row of a table or a column and performs a calculation for each row, then sums up the results.
   * **In your case**: It's used to sum the "Country Sales" values for each row produced by the WINDOW function.
2. **WINDOW**:
   * **Purpose**: WINDOW is not a valid DAX function, so let's assume you intended something like EARLIER, FILTER, or similar functionality. In standard DAX, you would use a combination of CALCULATE, FILTER, and ALLSELECTED instead to create a similar logic to a "window".
   * **What it's meant to do**: The goal seems to be creating a table of sales by country, and then calculating the cumulative sales based on some condition (like sorting or filtering).
   * **In your case**: The WINDOW function, as described, might represent a sorted list of countries, but the equivalent functionality in DAX would likely be handled by CALCULATE and FILTER.
3. **SUMMARIZE**:
   * **Purpose**: SUMMARIZE is a function used to create a new table based on some aggregation (like summing or grouping).
   * **In your case**: It creates a summary table where you have:
     + 'Main Data'[Country] (groups the data by country)
     + "Country Sales" (calculates the total sales for each country using SUM('Main Data'[Sales]))

So, this creates a table of countries and their total sales.

1. **ALLSELECTED('Main Data'[Country])**:
   * **Purpose**: ALLSELECTED removes any filters that may have been applied to the Country column, but still respects any selections made by the user in the report (e.g., if a user selected specific countries in a slicer).
   * **In your case**: It's ensuring that your calculation is done over all countries that are currently selected in the report. Without this, your calculation might be limited to just the current context of the report (e.g., just a subset of countries if a filter is applied).
2. **ORDERBY([Country Sales], DESC)**:
   * **Purpose**: This is used to sort the summarized data by the total sales of each country in descending order (largest sales first).
   * **In your case**: You want to sort the countries by their sales in descending order to calculate cumulative sales from the top country down.
3. **[Country Sales]**:
   * **Purpose**: This is the value that you want to sum up, after sorting the countries by their sales.
   * **In your case**: This represents the total sales for each country, which will be summed in SUMX.

**Recap:**

This formula is attempting to:

1. Create a table that summarizes the sales by country.
2. Sort the countries by sales in descending order.
3. Calculate the cumulative sales by looping over each country using SUMX

**Steps for Creating and using Parameter**

1. Define the Parameter to accept value of N - Create Visualisation - Create Ranking using RankX functions
2. Create Parameter from Modelling Menu - New Parameter - Name the parameter as Top Customer Count - it creates the table as per the parameter name
3. Configuration of the slicer to connect to the visualisation - Create measure as Display This Customer - Use this Measure in the Filter as Show when the Value is Yes"

1. Create Visualisation

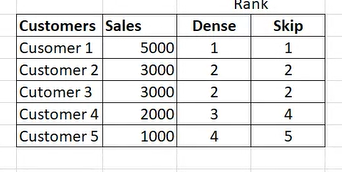
2. Create Ranking measure

3. Create New Parameter

4. Create measure for Display

5. Add Display measure in Filter

**Difference between dense rank and skip rank**



BINS IN POWERBI:

In Power BI Desktop, you can group data points to help you more clearly view, analyze, and explore data and trends in your visuals. You can also define the bin size to put values into equally sized groups that better enable you to visualize data in meaningful ways. This action is called binning.

If someone asks ki 0 to 10000 rupees madhe kiti lokanni sales kela, then it is known as bin, that range is known as bin.

Create a new table.

Size of Bins and Number of bins Difference:

Jevdha bin size motha thevdha data it can accommodate.