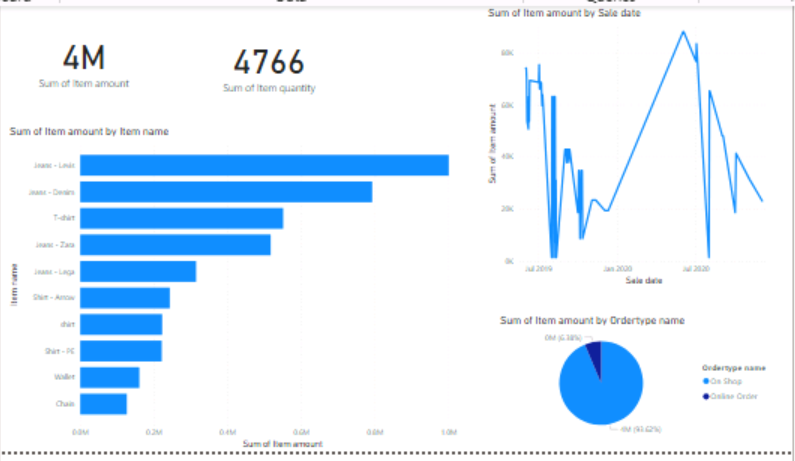
**Power BI**

**Power BI:** -Power BI is a business intelligence reporting tool which is developed by Microsoft to create reports and dashboards by using the tools in it. Like power query, power pivot, power view etc.

**Power BI desktop: -** Power BI Desktop used to create reports and it is free. We can connect to various data sources like SQL Server Database through Power BI desktop. We can do ETL using Power Query in power BI Desktop. We can do modeling in Power BI desktop in model tab. We can write Dax functions and we can see data in data tab, and we can create reports in report tab.

**Download: -** we can download Power BI from google or from Microsoft store if we download from google our Power BI won’t updates automatically but when we download form Microsoft edge our Power BI updates automatically.

**Creating First Dashboard: -** First get data from Get data and click on CSV/Text and select file click on file to see preview click on load. In report tab we create reports in report tab we have area to create report and filter plane where we can apply and see the filters. Visual plane in which we have visuals used to create dashboard. Data plane or Field plane it has Tables, Measures, Columns of data.



**Focus Mode**: - it shows expands that visual to whole page click back to back to report.

**Filters: -** Filters plane present in report tab. There are 3 types of filters filter on Visual, Filter page, Filter on report. Filters are used to filter data in visuals.

**Power Query: -** Power Query is an ETL tool which extract data from source and performs various operation on it to transform and load data to destination (Power BI Desktop). Power query has interface which helps us to do transformation, or we can write M Query to do transformation. When we use interface also Power Query uses M Query in background to perform transformation.

**Power Pivot: -** Power Pivot is the place where we do modeling our data / creating relationships between data. Data view and Model view comes under Power Pivot. We use DAX in Power Pivot.

**Power View:** - Power View is the place where we create visuals. Report view is nothing but Power View.

**Note: -**We spend more than 80 percent of time in Power pivot and Power Query.

**Power BI Desktop =Power Pivot + Power View + Power Query**

**Data Types: -** Following are the data types in Power BI

Text – Customer Name

Whole Number – 1,2,3,4 etc.

Decimal Number – 1.23,2.345 etc.

Boolean – True/False

Date Time – 12/02/2021 02:30:33

Date – 12/02/2022

Time – 02:20:20

We need to give appropriate datatype to avoid data loss and to perform operations correctly.

We can change datatype by click on column goes to column tools change datatype. Numerical datatypes have sigma before there column names.

Default aggregation of numerical value is sum we can change by click on down arrow after adding column to fields shelf.

Default aggregation of text datatype is First. We can change aggregation of text datatypes same as numerical value datatypes.

Default aggregation of date and time is earliest we can change aggregation of time and date same as numerical value datatypes.

**Formatting: -** Formatting used to make your report looks good and attractive. We have general formatting and visual based formatting. General formatting is same for all visuals it has tittle, effects, background, alt text [which display when visual not loaded] etc. visual based formatting may have data labels [display numbers on bars], we can change bar colors, plot area background, x-axis, y-axis, we can change value representation and orientation etc. we can change font and color of values and text.

**Get Data: -** Get data used to import data into Power BI. we can connect to different sources using get data like Excel, Folder, CSV, Database, Website, Azure etc. we can navigate to Power query by click on transform when we are getting data into power BI desktop. We can load data into Power BI desktop by click on load. We can see preview by click on table. Click check boxes of tables which we want to load. While loading data it shows datatype detection by first 200 rows change it to entire dataset to avoid getting error.

**Interaction Between charts:** - There are 3 types of interaction between charts.

Highlight, Filter, None

Highlight- it highlights part of data when user click a particular segment in other chart or visuals.

Filter- it redrawn for the date of the data point which selected in other visual or chart.

None- it shows no action on visual when we select other visuals or charts. i.e. no interaction.

To edit between intersection’s use edit interaction in format tab [shows when we click on visual] then we can see 3 icons for filter, highlight, none for every visual at top. For some visuals like card, muti row card have filter and none no highlight.

Default interaction between visuals in highlight.

**Hierarchy:** - Hierarchy is a way of arranging of data in organized way at different levels based on their granularity and relation for better understanding and accessing.

When we are using hierarchy in our visuals then 4 icons are displayed

Drill up- goes to top level hierarchy.

Drill Down- goes to deeper level of hierarchy

Next level hierarchy -goes to next level like data shows at year level we click on it is shows data in quarters like Q1, Q2, Q3, Q4 not like quarter breakdown in each year.

Expand next level hierarchy- expands to next level like year level to quarter breakdown in each year.

To create hierarchy, click on column click on 3 dots click create hierarchy. Now click on other column which we want to add into hierarchy select 3 dots click on add to hierarchy or we can simply drag and place all columns in hierarchy.

By adding column one upon one in chart selves also forms temporary hierarchy for that visual only.

**Drill Down and Drill Up: -** Drill down used to goes to deeper level in hierarchy and drill up used to go top level in hierarchy.

**Drill Through: -** Drill through helps us to explore related data in more detailed way. By using drill through we can navigate from one page from other page and even from one report to other report where can see related data in more detailed way at deeper level.

To drill through to other page place field in drill through selves under in visual plane in detailed page.

Now click on column on chart and click or visual drill through then we navigate to detailed page where we can see more related data to that selected data point. We can add multiple columns in drill through.

**Keep Filters:** - it keeps all filter applied on main page and applies to detailed page. If we of it filters on main page does not reflects on detailed page.

**Tool Tip: -** Tool tip used to provide some extra information about the data point on which we hover . we can add new columns in tool tip selves of visual to see more information and we can also create a custom tool tip to show visuals based on data point we hover. To do that follow following steps

1.create new page and set it type as tool tip to do so click on page go to page formatting in canvas setting select page type as tool tip

2.add visuals we want must related to fields in main page.

3.Go to main page click on visual general formatting click on tool tip and select report page.

Now we can see dynamic tool tip with visuals when we hover on data point.

**Custom Visuals:** - Custom visuals are the visuals which by default does not present in visual plane we need to import them there are two methods to import: -

click on 3 dots in visual plane click on get more visuals than we get all visuals, and we can import that visual into our report.

Go to Microsoft app store go to power bi visuals download them and click 3 dots in visual plane and select import visual from file.

These custom visuals are report specific if we import a visual which is not going to present on other report.

**Actions: -** Actions are the special formatting option which can be used for buttons, shapes and images there are different types of actions are there in Power BI.

**Page Navigation-** it used to navigate from one page to other click on visual action page navigation select destination page.

**Back-**it used to get back to previous page click on visual action back.

**Web URL**- it used to navigate to specified web URL click on visual action select Web URL and give URL in URL selves.

**Drill through** - it used to navigate from main page to drill through page. Visual action Drill through select destination page. Now click on drill through column and CTR+click on button.

**QA-** it used to query our dataset in natural processing language and we can get insights from it.

Do CTR+Click on buttons to perform their respective action.

**PQ-Remove Rows: -** used to remove rows from the table.

Remove Top Rows- specify numbers of rows to remove.

Remove Bottom Rows- specify numbers of rows to remove.

Remove Alternative Rows- start row, number of rows to remove number of rows to keep. It does remove reparative up to last row of data.

Remove Duplicate Rows- removes duplicate rows. First row of that value is kept repetitive value rows are removed.

Remove Blank rows- removes blank rows.

Remove Error rows- removes error rows.

**PQ-Fill: -** It used to fill the gaps in the source data. In Fill we have Fill up and Fill Down

Fill up fill the down data to empty cells above it.

Fill down fill top data to empty cells below it.

**PQ-Manage Columns:** - To manage column we have Move, Choose Column, Remove Column.

Move- used to move columns left, right, at beginning and at end.

Choose columns- we can select column which we want to kept in our model remaining columns get deleted.

Go to column- it used to navigate to that column.

Remove column used to remove selected column

Remove other columns used to remove columns which are not selected.

**PQ-Split Column:** - used to split column. We can split a column in PQ in following ways.

**Transform-Split Column**

By delimiter – we need to specify delimiter.

By number of characters – we need to specify number of characters.

For above two we need specify that each occurrence, left most or right most.

By positions- we need specify the position at which split should occur values must be ascending order.

For above 3 we need to mention split into row or column.

Lower to upper, upper to lower, non-digit to digit, digit to non-digit this four are split column based on transition.

**PQ-Derived Column: -** we can create ne column in PQ in two ways.

Using custom column in add column tab and use m language in it to create.

Using conditional column in add column tab there where we can have if else we can have multiple if else also. Create column based on condition.

**PQ-Append:** - it used to append two or more tables it like union all SQL after append file have duplicated to remove duplicates select all columns and right click select remove duplicates or select all columns click on remove rows click remove duplicates.

To append two tables, use basic to append more than two tables use advance and select all tables.

Append performed by matching name of the columns not by position.

Append query add new query to existing query and append query as new create a new query by appending both means we have query1,query2 and merged query of Q1 and Q2.

**PQ-How to calculate Age from Date: -** To calculate age from date go to power query select your query and select date column may be date of birth click on date click on age in add column or transform [ add column adds new column where transform change existing column]. Now we get result in duration change in to total years by click on duration in transform click total years.

**PQ-Group By: -** it is same as group by in SQL server it used to aggregate the data at specified level or at high level.

Use basic to group by one column and one aggregate function.

Use advance to aggregate based on multiple columns and have multiple aggregation.

We can do Sum, Min, Max, Avg, Count, Median, count distinct, All rows means it kept whole table for that value for example we do aggregate by location with all rows now a table created with all rows related to each region. If we want to see data expand that tables.

Location1 -table of data related to that location1.

Location2 -table of data related to that location2.

**Measure Table: -** Measure table is table which consist of only measures. Measure table always helps to find measure easily because we kept all measure in model in one table.

To create measure table create a table with enter data and delete rows in it and move all measure to the measure table by click on measure and change home table.

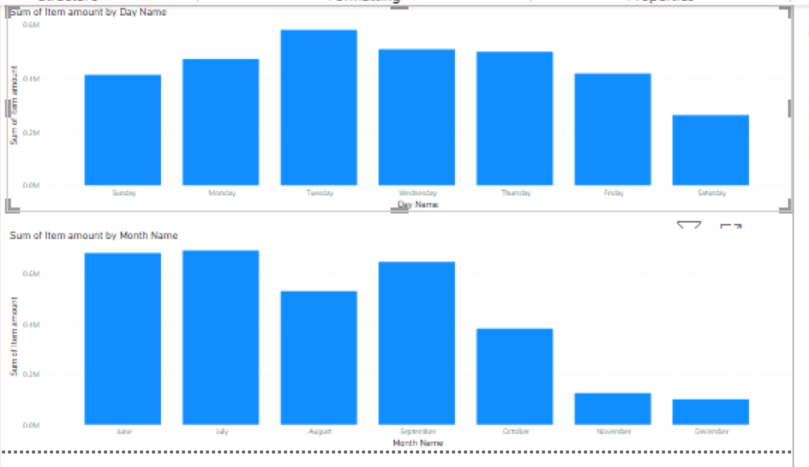
**PQ-Merge Tables: -** Merge tables means joining tables similar like using joins in SQL. We can perform right, left, right anti, inner, full, left anti joins.

Click on any query then click merge queries if we want to join query to existing query click merge queries if we want to create new query then click merge query as new now select two tables and select column on which merge going to happen if multiple columns ctrl+click and select type of join and if we want to enable fuzzy matching check fuzzy matching and set it settings below.

**PQ-Sales/Weekday and Month name: -** to get sales based on weekday click date column go to add column click date the day next day of week we get a column with day of week name and to get month name click on date column go to add column date click on month next month name.

Now we encounter a problem that day names and month names are sorted either on apathetical order or sum of sales but we need them in chronological order like Sunday, Monday... and January, February… then we need to create two columns one is day of week and month number we can get both by add column date use month and month for month number and add column date day next day of week to get day number in a week.

Now sort Weekday by day of week by click on weekday click on sort there we can sort one column by values in other column now we sort weekday with day of week and month name with month. Now result look likes below: -



**Lists and Bins:** - Both Lists and Bins are used to create groups in Power BI. Click on column select new groups.

Select lists in group type and the we see all distinct values in the column and select values and click group and give name to group we can create as many groups as we want with values, values which are not in groups are shown under other groups.

Select Bins in group type then we need to select bin type either as size of bin or number of bins. We can see max and min value in column. When we are selecting number of bins also size of bins

Main difference between bin and list is we can hand pick values in list where in bin we need to set size or number of bins.

**Building relationships: -** we get data from different tables we need to establish relation between tables to work with both tables data to create a well-designed and optimize report. There are 3 types of relations one to many, many to many, one to one.

We build relations in model tab.

Click on manage relationship click new or detect if Power bi not detects automatically click new select tables and column between them.

Cardinality means relation between to tables like many to many, one to one, one to many.

Cross filter direction always from one side to many if it is one to one or many to many then cross filter direction is both.

Assume referential integrity by check this Power BI always does inner join rather than outer joins.

Apply security filters on both sides used when we use RLS on tables having many to many cardinality of cross filter is both.

In mange relation we can edit and delete existing relationships.

We can forcibly set the cross filter in both direction but if effects model performance.

Most of the time Power BI automatically detects relation and cardinality between tables.

**Fact Table:** -Fact table is a table which consist of facts/measures it consists of primary key and multiple foreign keys which always present at center of schema. Fact tables are generally large and updates very frequently.

**Dimension Table: -** Dimension tables is the table which consist of information related to the facts/measures in fact table they are generally small and updates rarely. Mostly they consist of primary key no foreign keys some time one foreign key.

**Schema: -** Schema is the logical representation of objects/ tables in a database.

There are two types of schemas.

**Start schema** in this schema fact table in center and dimension tables are directly connected to fact table.

**Snowflake schema** is this schema fact table is at center and consist of dimension and sub dimension tables also present.

Start schema is less complex and denormalized when compared to snowflake schema.

**PQ-Unpivot:** - Unpivot used to reshape the table it converts columns into distinct values in the row. Columns which undergo unpivot are converted into attribute and value pair. Attribute has column names.

We have 3 options unpivot columns, unpivot other columns, unpivot select columns.

Unpivot columns it unpivots selected column.

Unpivot other columns it unpivots all other columns which are not selected.

Unpivot selected columns it unpivots selected columns.

Here the main difference between unpivot columns and unpivot selected columns is in unpivot columns if any new columns added to data that also unpivots unpivot other m query used by PQ where in unpivoted selected columns new column remains unchanged because it is not selected here unpivot m query used by PQ.

**Date Dimension: -** usually Power BI automatically creates Date Dimension for each date column so model become complex and increases the size and by using date dimension we can do analysis at week, weekday etc. as our requirement we can create columns and we can perform analysis.

Go to setting options in local and global uncheck automatic date/time to avoid creating date dimension by power by for each date column.

We can create date dimension in 3 ways:-

We can get all dates from a excel.

We create a date tables by using DAX using Calender(start\_date,End\_Date) and by using CalenderAuto([end month of year]) if we give 3 year starts from 4 month.

We can create by using M query. Click on new query click on blank query and write following m query.

=List.Dates(#date (year, month, day), count, #duration(day, hour, minute, second))

=List.Dates(#date (2019,01,01),3650, #duration(1,0,0,0)) – dates for 3650 days.

**Top (N)/Bottom (N):** - Top N filter used to filter the top records based on a column values.

Let see in this case we need top 3 products based on sales.

Create a chart with item sales and item amount.

Click on visual now go to filter plane click item name because we want to see top 3 names add item amount in filter selves select top and 3 in box. Click on now we can see top 3 products.

We can select bottom in place of top as we needed.

**Canvas Setting:** - we can set the size of canvas(page) by using canvas setting. Click on page go to format click on canvas setting there change page size to tool tip, 16:9[default], 4:3, letter and custom. In custom we can set height and width of page.

**Wallpaper: -** Wallpaper is larger than page background and we can say wallpaper consist of page background wallpaper is superset of page background. Click on page go to format there we have some options on wallpaper like color, transparency, image etc.

**PQ-Find Occurrence of a character: -**

Create a custom column in add column replace a with “” by following M code.

Text.Replace([Column\_Name],”Character”,””) use double quotes to represent strings.

Create other custom column in add column by following M code.

Text.Length([Column\_Name])-Text.Length([Replace\_Column]) then that result equivalent to occurrence of that character.

**Change Datasource: -**

Click on transform data, data source setting click on change data source and browse for new source which we want to change and click ok. Now visuals in our report redrawn for values in new file until and unless both fields have same metadata. We can use it as incremental loading but we need to doing it manually.

**Enable search in slicer: -**

Slicer is a visual in power BI used to filter the visuals in the page with selection of user.

In filter plane also we have search but it used search columns which are involved in filter where in slicer search used to search values in the column which we using in slicer to filter the visuals in the page.

Click on 3 dots and click on search which enables us to search the values in slicer.

**Drill Through Button: -**

Drill through means navigating from high level page to detailed page.

We can create drill through button by using drill through action.

Create visuals in both high level and detailed page place suitable column in drill through selves in detailed page.

In high level page insert button and set action as drill trough and select destination page as detailed page.

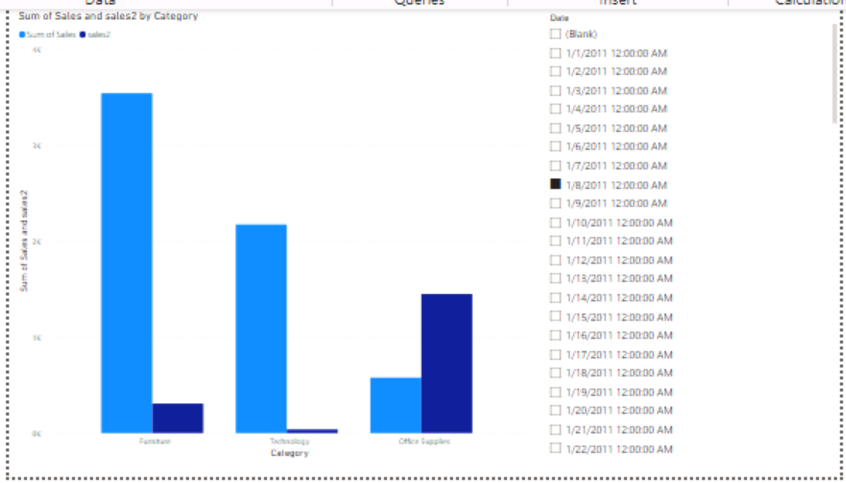
Now we click on datapoints in high level page and Ctrl+click on button it navigates us to detailed page where all visuals shows related data to that selected datapoint.

**Role Playing dimension: -**

Role playing dimension is the dimension which performs multiple roles like if we have sale date and ship date when we connect both dates to date field in date table now date dimension is a role-playing dimension because date in date dimension acts as sale date and ship date but in Power BI only one active relation is possible for this we use userelation Dax function for inactive relation. In userelation we specify the column in sales table and date column in date table

Calculate(Sum(Sales[sales]),userelation(sale[ship date],date[date]))

We use multiple Date dimensions as alternative to role playing dimension but it makes model more complex. In below we can see that we are calculating sales for same date one sales represents sum of sales for order date and other for ship date.



**Bookmarks: -**

Bookmarks capture the current state of the reports with filter and slicer selections which can used to navigate as story or presentation.

To create bookmarks, use following steps.

Create a report with two types of visuals in it for country and category.

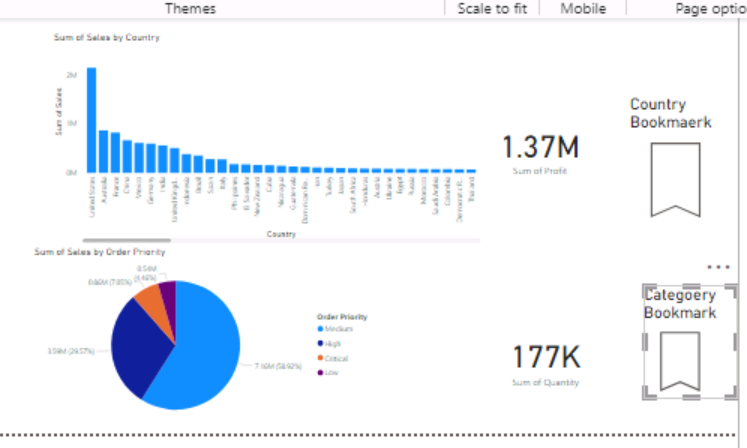
Click on view click on bookmarks add two bookmarks country and category and click selection plane.

In selection plane select visuals which you want to display and which you want to hide and click 3 dots at bookmark name click update.

Now add bookmark buttons and add bookmarks to it in actions. Now when we ctrl+click on bookmark buttons we get corresponding visuals.

A screenshot of a graph

Description automatically generated



**Creating Folders: -**

Folders helps us to organize the data. To create folder, go to model Tab click on column then we see the properties window in that window we have a selves to write folder name.

For columns which want to be under same folder write same name to all columns in folder name avoid typo error because typo error creates new folders.

**DAX: -**

DAX means data analysis expression. It used to analysis the data.

DAX is a functional language it consists of functions.

DAX consists of functions and operators which helps to build the formulas and expressions in Microsoft BI tools.

DAX used in Power BI, Power Pivot in Excel, Azure Analysis Service, SSAS tabular model.

We can use DAX in power BI in Calculated Columns, Measures, Calculated Table.

**Calculated Column v/s Measure: -**

Calculated Columns are the columns which are derived from existing columns using DAX.

Measures are the single summarized values. DAX used to created the measures.

|  |  |
| --- | --- |
| **Calculated Column** | **Measure** |
| It uses Row context | It used filter context |
| It increases the size of the file | It does not affect the size of file. |
| Less analytics | Rich analytics |
| They created before run time. | Evaluates at run time. |

**Row Context and Filter Context: -**

Row Context calculated for each row within the values of that row.

Calculated columns are example of row context.

E.g. if we have product and category we need product-category use calculated column because we need this value for each row.

Filter Context calculated by evaluating all filters applied on data.

Measures are the exam of filter context.

E.g. if we want sum(sales) for a particular year, month or based on any other filter use measure.

**Enter Data: -**

We can see enter data in Power Query and Power BI desktop. It used to store the data.

But use Enter data to store static data which does not changes.

If we paste our data in enter data if any changes in the source, they won’t reflect in Power BI so use data which is very rarely changes.

**Calculate: -** Calculate evaluates the expression in the modified filter context. We can specify the filter in calculate function and evaluates the expression in it based on that filter.

We can specify the Boolean expressions, table expressions like filter and filter functions like all, allexcept etc.

Calculate (Expression, <Filter1>,<Filter2>,….)

**SUM and SUMX:** - Sum is a DAX function it is same as Sum in SQL and SUMX is a iteration function which perform sum operation by iterating each row.

For example, we have quantity and cost of product so we cannot use sum because sum(quantity) \* sum(cost) does not gives result. By using SUMX it iterates through each row and perform quantity \* cost.

Sum (Column)

SUMX (Table, Expression)

**Variables: -** Variables used in DAX to increase readability and reduce line of code. We declare Variables by using keyword Var and use return to return the result.

By using Variables

variable =

var lastyear=CALCULATE(sum(Data[Sales Amount]),SAMEPERIODLASTYEAR(Data[Sale date]))

var sales=sum(Data[Sales Amount])

var res=Divide(sales-lastyear,lastyear)\*100

return res

Without using variables

with\_Out\_Varaible =Divide(sum(Data[Sales Amount])- CALCULATE(sum(Data[Sales Amount]),SAMEPERIODLASTYEAR(Data[Sale date])), CALCULATE(sum(Data[Sales Amount]),SAMEPERIODLASTYEAR(Data[Sale date])))\*100

By using variables we can debug easily.

**Disconnected Dimension: -** Disconnected dimension is the dimension, which is not connected to our model, but we are using in our model.

We know in Power BI when we click on date, we see data at that date but if we want to see total sales up to that date, we need to use disconnected dimension. In following DAX, we declare a variable and assign the value we selected in date slicer in calculate function we calculating sales up to that date using order\_date<=selected\_date.

Measure = var d = max(date\_dim\_dis[Date])

 return CALCULATE(sum(DAta[Sales]),DAta[Order Date]<=d)

**Display Custom Message:** - some time there is no data for selected filter in that case showing blank is no look good so we can display a custom message.

Now take a card and use following DAX function in it. Make border off and background off to card.

Here [s] is a measure sum of sales.

m = IF(ISBLANK([S]),"There is no data or selected month please select other month","")

if value is black it goes to true and display message if it false it did not display anything.

**Distinct: -**

Distinct used to return the distinct values from the specified column or specified table. But we can’t pass multiple columns.

Distinct is a table function it returns a table with one or more columns based on we pass column or table. If we pass column we get one column if we pass table we get multiple columns [columns in that table we passed].

**Distinct(column/table)**

**All, Allexcept, Allselected: -**

All, Allexpect and Allselected are filter functions which we use in calculate function to change the filter actions on that measure.

**All** it avoids the all filters applies on it whether it is a table or column if it table any filter on that variable will not effect the measure if we specify column any filter on that column will not affect the measure.

All(Table\_Name or Column\_Name)

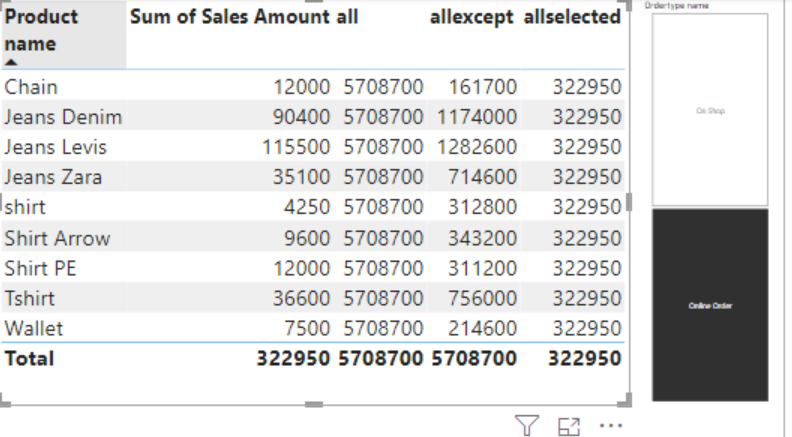
**Allexcept** it avoids all filters on table except specified one for example we use all except of product name only filters on that column will affects the measure.

AllExpect(Table\_Name,Column\_Name)

**Allselected** it avoids all internal filters[product name I each row is a filter] and changes for all explicit filters.

Allselected(Column\_Name)

We use above 3 functions in filter of calculated function.



**TotalYTD, TotalQTD, TotalMTD: -**

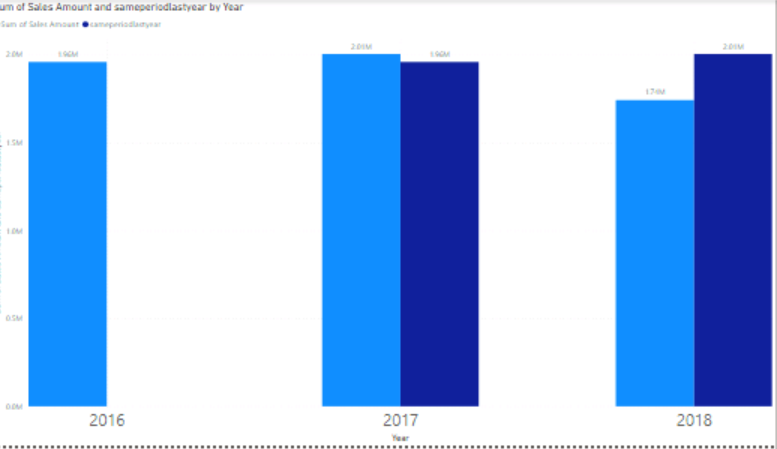
TotalYTD, TotalQTD, TotalMTD are used to calculate running total. Where TotalYTD calculate running total per year, it holds the previous data step by step up to year, TotalQTD calculated running total per quarter, it holds the previous data step by step up to quarter and TotalMTD calculates running total per month, it holds the previous data step by step up to month. We can specify a filter condition only one filter.day/month defines fiscal year starts from.

TotalYTD( Expression,Date\_Column,[filter],”month/day”)

TotalQTD( Expression,Date\_Column,[filter],”month/day”)

TotalMTD( Expression,Date\_Column,[filter],”month/day”)

**SamePeriodLastYear: -** it returns same period of previous year. We use it in calculate function to return same period and perform expressions on top of it. Drill down also works on it if we drill down from year to quarter it shows previous year same quarter sales with current quarter if we drill down to month it shows previous year January to current year January.



**Dynamic Tittle: -**

Tittle should change based of selections we made on charts like if we drill down to quarter it shows which year that quarters belongs to same way if we drill down to quarter is show year and quarter which they belong to same for month. We can achieve it by following day function.

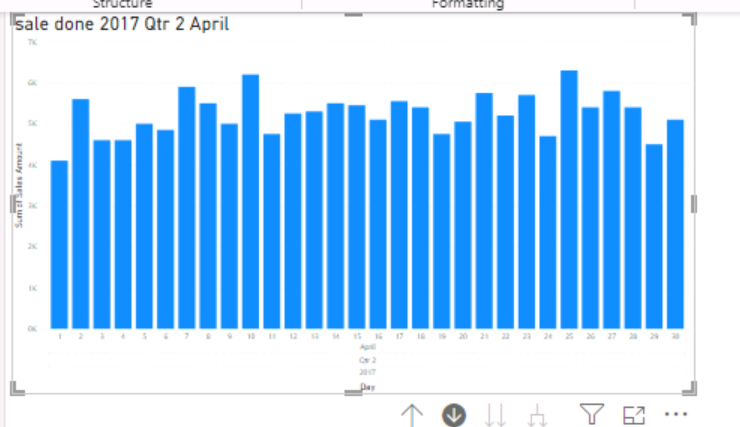
Dynamic Tittle=

Var year=IF(IsFiltered(Data[Sale Date].[Year]),SelectedValue(Data[Sale Date].[Year]),” By Year ”)

Var quarter=IF(IsFiltered(Data[Sale Date].[Quarter]),SelectedValue(Data[Sale Date].[Quarter]),” ”)

Var month=IF(IsFiltered(Data[Sale Date].[Month]),SelectedValue(Data[Sale Date].[Month]),” ”)

Return “Sales done” & “ “ &year&” “&quarter&” “&month



**Related:** - Related DAX function used to get the column from related tables. We related at many sides of table to get related column from one side of table. When we write related automatically all related tables and their columns will display.

**Related(table[column])**

**Remove Filters: -** remove filters is same as all we can say it as alias to all. Both have following differences.

In direct query mode we cannot use remove filters use all in that case.

We can use all as individually function but for Removefilter we can use it as filter in calculate we cannot use it individually.

When we use all ALL as individual function it removes all filters from the table and returns the table but when we use it in filter in calculate it removes filters from table but not return as table so to avoid this, we use remove filter which works same as all when it used individually.

**DateMTD, DateQTD, DateYTD: -**

DateMTD works same as TotalMTD, DateQTD works same as TotalQTD and DateYTD works same as TotalYTD. Following are the main difference between them.

TotalMTD, TotalQTD, TotalYTD used as individual functions, where DateMTD, DateQTD, DateYTD used in filter of calculated function and can use individually.

DateMTD, DateQTD, DateYTD returns table with dates respective MTD, QTD, YTD if it is MTD it returns dates for 30 days from current date in case quarter it returns 90days for year it returns 365 days. Where TotalMTD, TotalQTD, TotalYTD returns scalar value.

We can use only one filter in TotalMTD, TotalQTD, TotalYTD but in DateMTD, DateQTD, DateYTD we can use multiple filters. Month/day used to represent fiscal year.

DateYTD(Dates,[filter1],[filter2]..,[”month/day”])

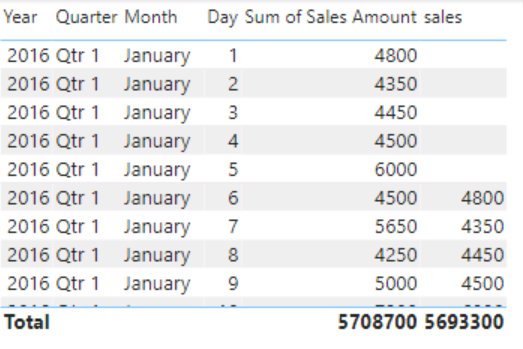
DateQTD(Dates,[filter1],[filter2]..,[”month/day”])

DateMTD(Dates,[filter1],[filter2]..,[”month/day”])

Parameters in [] are optional.

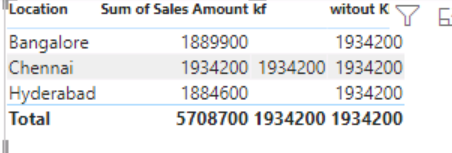
**DateADD: -** DateADD used to add particular no of intervals to date. By using sameperiodlastyear we can get data for same period in last year but by using date add we can get data for any interval like 3 days back, 4 years before etc. to get upcoming data use positive number in interval to get past data use negative number in interval.

DateADD(Date, no of intervals, interval) intervals means day, month, quarter, year.

In below we comparing previous 5th day sales with current sales.

**KeepFilter: -** by using keep filter we can modify the filter how it applied while evaluating the expression in calculate function.

For example, we want to see sales for Chennai then we create a filter with out using keep filter now we place in a table with all location then we see that measure shows same value i.e. Chennai sales for all other locations when we use keepFilter it shows sales for Chennai location only remaining it shows blank see below figure.



**Split values into rows: -**

Normally we split values into column by using split column to split them into rows click on split column select any option there like split b delimiter, split by positions, split by characters etc. and in **advanced options select split by rows.**

**Import VS Direct Query: -**

Import Query imports the copy of data into Power BI model. Visuals query the data from that data. Data automatically does not refresh we need to do manually or do schedule refreshes. But by import mode we can use all DAX queries and import mode is fast compared to Direct query. Import query not much suitable for large data. Because in power BI we have size limitation.

Direct query does not copy data into power bi it just imports data about data mean metadata visual query the data directly from underlying dataset at source it always give most recent data. But we are unable to use some DAX function while we are querying in direct query. We can eliminate file size constraint by using direct query and no need to schedule refresh.

max size of dataset is 1gb and max file size is 10gb with premium.

**Union, Insert, Except: -**

Union gives combination of all tables it gives duplicate rows also to remove duplicate rows use distinct function.

Intersect give common columns between both tables.

Except it gives rows which are present in first table and not in second table.

For above all functions both tables must have same number of columns and have same name and columns must in same position. Same as SQL Server.

**Handling Nulls and Blanks: -**

Blanks and Nulls are not same Null means no data where Blank means empty it many contain tab, space etc. so first we need to replace blanks into null to do so first trim that column which have blank cells to avoid all spaces and tabs next click on replace values replace empty cells with null.

Now when we are using M code be care full with null because nulls in M query return null so while writing M query take attention on nulls and take respective action on them like if value is null do this.

**Reference VS Duplicate: -**

Duplicate means the new duplicate table and original table does not have any relation between them. All steps applied on source are also copied to duplicated table.

Reference means then referenced table always linked with original table if any change occur in original table it automatically reflects in the reference table.

**Selected Value:** - Selected Value DAX function returns the selected value of a column in the slicer or in a visual. By using this selected value, we can use disconnected dimension. We can pass alternative value if no value is selected it display that alternative value.

SelectedValue(Column, Alternate Value)

**Hasonevalue: -** It returns true if specified column has one value for each type i.e. unique. We use this in case at product category level we want to see sales in product name level we want to see profit. When we drill down all values in product name repeated one time only.

val = if(HASONEVALUE(Data[Product name]),sum(Data[Profit]),sum(Data[Sales Amount]))

**Column From Examples:** - It used to derive a new column from a selected or all columns by passing input to derived row until Power BI finds pattern. Power BI writes M query itself based on pattern we want. Add Column Columns from Examples.

**Refresh Data:-** We can refresh single Table, Multiple Tables and All tables in power BI. We can do this refresh when we load data as import.

Refresh all go to Power Query (Transform Data) click on refresh preview click refresh all and in Report tab we have refresh option it refresh all tables.

Refresh Table: - It refreshes Single table for this in report vies click on 3 dots of at table click and click on refresh.

Refresh Multiple Tables: - It refreshes multiple tables by selecting multiple tables and click on 3 dots at table name and select refresh but in report tab we cannot select multiple Tables we can select multiple tables in Model Tab.

**Synonyms in Q/A: -** Q/A is used to query the data by natural language.

Synonyms used to train Q/A when user search profit as margin it does not give results so by adding synonym margin to profit if any one search as margin we get profit.

To add synonyms click go to model tab click on column in synonyms add synonym word to that column.

**Custom Themes: -** we have themes in view tab. There we have some default themes provided by the Power BI. We can custom that themes by click on custom themes and we can customize as needed and we can save this customized theme to use in other report for that in view tab under themes click on save current there it saves as Json document. Now while creating new report and want to use that saved theme click on view under themes browse theme click on Json file we saved. Now new report all pages have same theme.

**Data Preview: -** By using Data Preview, we can know the about the data in the column. There are mainly 3 types of tools in data preview.

Data Quality it displays percent of rows valid, error and empty.

Data distribution it displays how many distinct and unique values in the column.

Data Profile is combination of data Quality and Data Preview and shows some additional information like min, max, std, avg etc.

**Slicer VS Filter: -**

Filters are developer friendly user unable to access them where slicers are accessed by both users and developers.

Filter is more flexible than slicer because in filter we can do basic, top, and advance using and or with conditional operators like less than, equal greater then etc.

In filter we have filter on visual, page and report where this is not there in slicer.

We can achieve filter on visual, filter on page filter on report by using edit interactions and sync slicers.

**View over Tables: -**

When we connecting to database connect through view is best practice because of following.

Views acts as abstraction to tables.

We can perform complex joins in view instead getting tables into Power BI and doing here.

Views hides database-specific objects from users.

Views can used to help you to easily manage the dependent system. i.e. when we directly connected to table if column name changes it effects visuals in but when we connect to view we can use a alias name in it by this any change in column name does not effects model.

**Get Total along the axis: -**

To get total along the axis follow the following steps-

Duplicate the data.

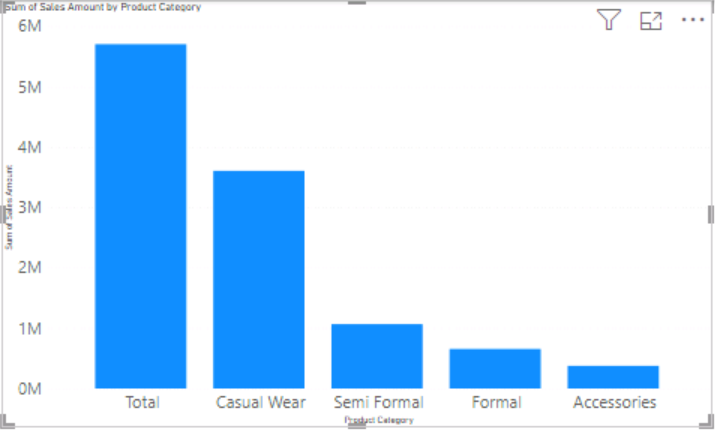
In new table group by date and sum the total sales for each day.

Now append new table with existing one.

Only two columns get matched and remaining shows nulls.

For numeric data type replace null with 0 and for non-numeric replace null with none.

For product category add total in place of null. Close and apply. Create bar chart we get result as follows.



**Calculating Running Total: -**

Create a measure by following DAX to create running total measure: -

running\_total = CALCULATE(sum('Data (2)'[Sales Amount]),'Data (2)'[Sale date]<=MAX('Data (2)'[Sale date]))

if case we don’t have sales continuously, we same running total for that days to avoid it use following Dax.

running\_total = if(SUM('Data (2)'[Sales Amount])>0,CALCULATE(sum('Data (2)'[Sales Amount]),'Data (2)'[Sale date]<=MAX('Data (2)'[Sale date])),blank())

**Formatting Header Icons: -**

In general tab we have header icons where we can manage header icons like adding icons, removing icons and formatting them like adding background, border, colors etc.

**PQ-Pivot Table: -**

Pivot table used to pivot the table pivot means converting distinct values in a row into columns and we can select the value field with aggregation.

**PreviousMonth, PreviousYear, PreviousQuarter, PreviousDay: -**

This DAX formulas are used to calculate previous month, day, quarter, year sales corresponding to current date.

Specify dates column as parameter. No drill through possible.