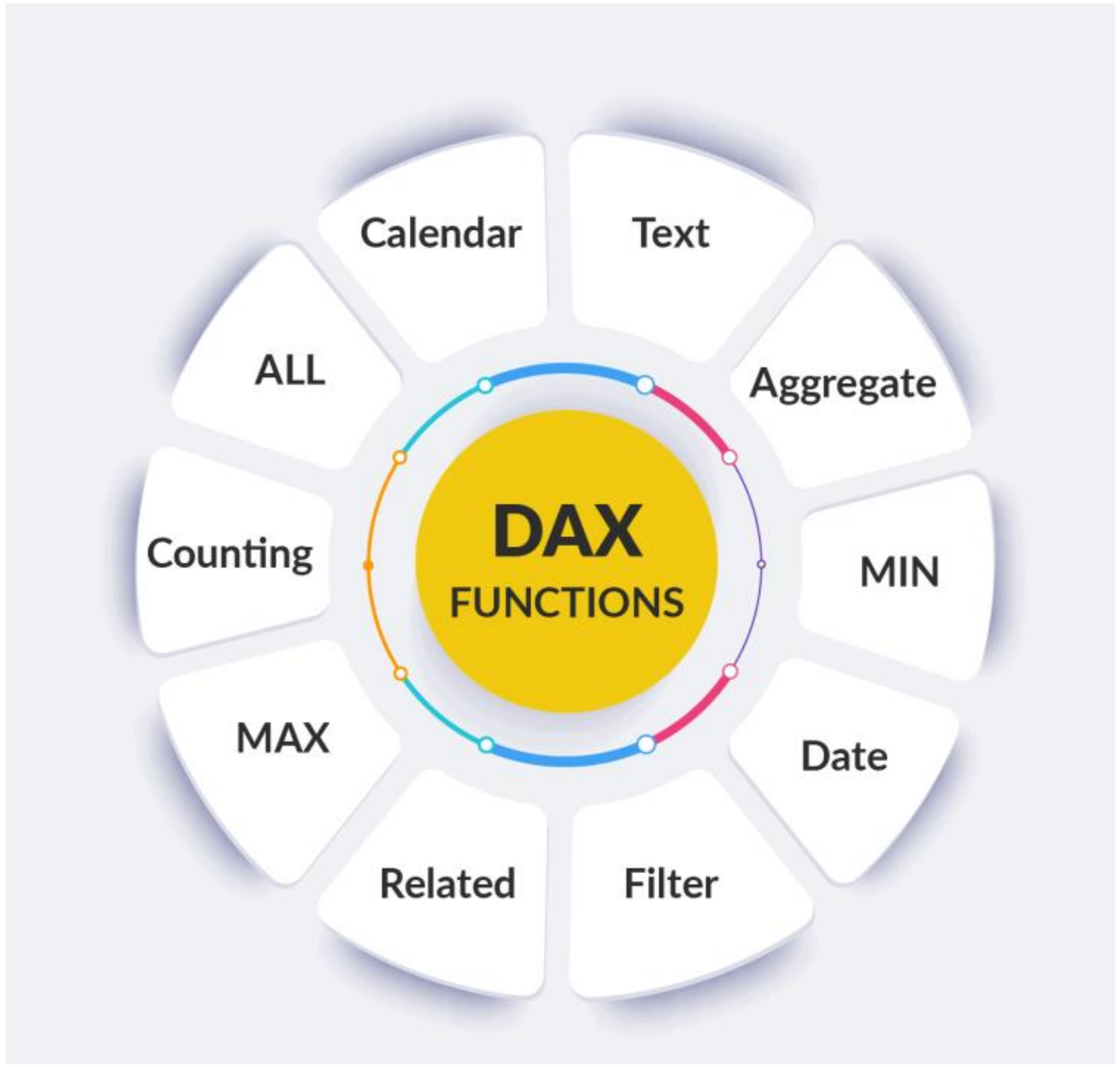
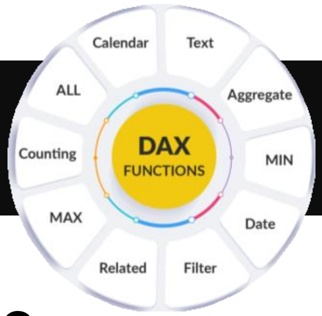
DAX Formulas in Power BI

SUM FORMULA


DAX
FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

SUM Formula in Power BI



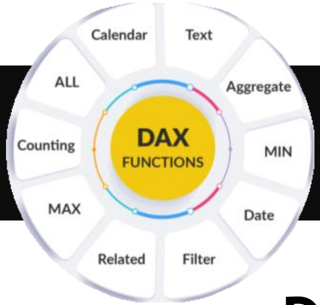
Definition: SUM is a simple aggregation function that adds up all the values in a single column.

Example

Total_Sales = SUM(SalesData[Sales])

Use Case: Use SUM when you want to sum up all the values in a numeric column without any complex row-by-row operations.

Performance: Since it operates directly on a column, it's fast and efficient for simple summations.



Calculated Column in Power BI

Definition: A calculated column is a new column that you create in a table using DAX (Data Analysis Expressions). The value for each row is calculated when the column is created, and it remains static unless the data is refreshed.

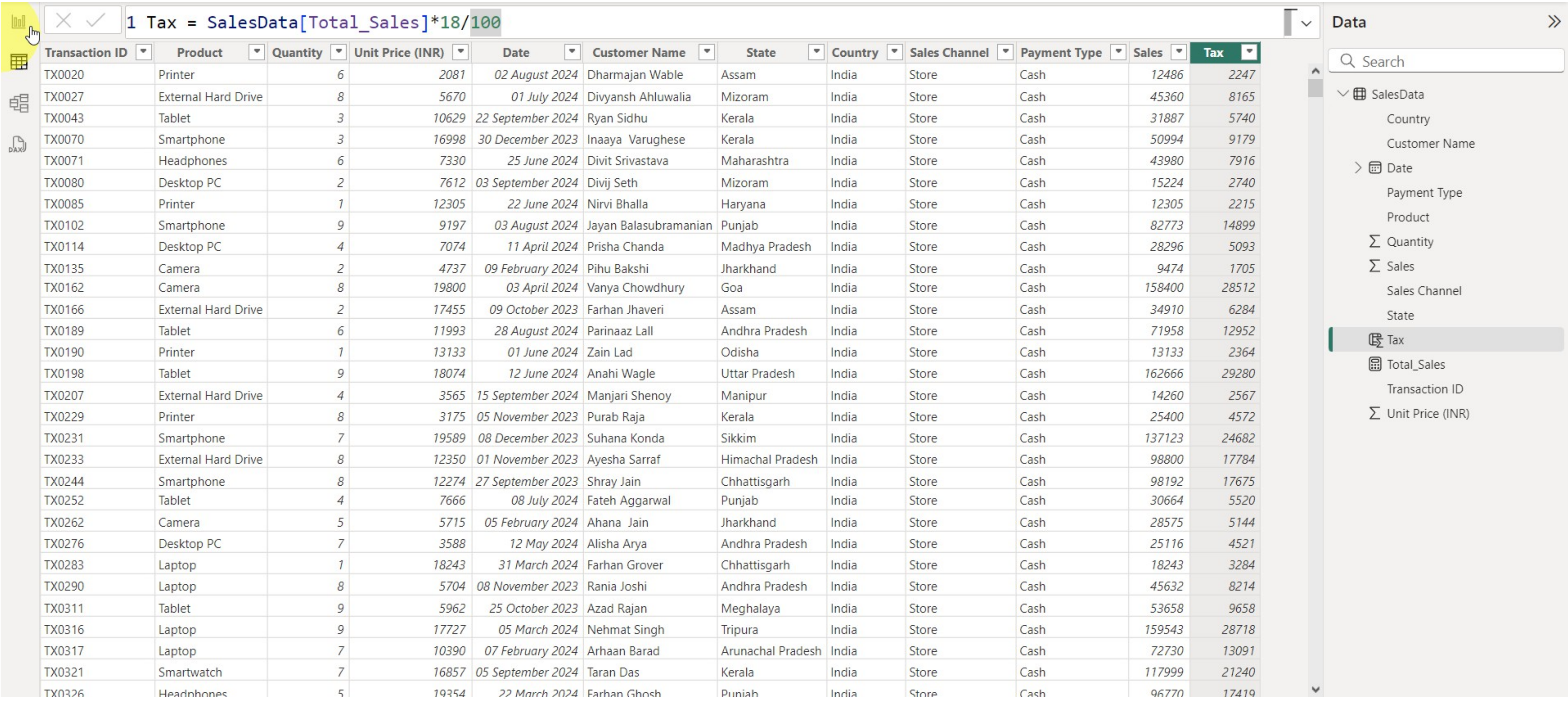
Example

```
Tax = SalesData[Total_Sales]*18/100
```

Use Cases: Suitable for scenarios where you need a value for each row in the table, such as adding a new field derived from existing data (e.g., creating a "Total Price" column by multiplying "Quantity" and "Price").

Performance: Since calculated columns are stored in the model, they can affect performance, especially if the model is large.

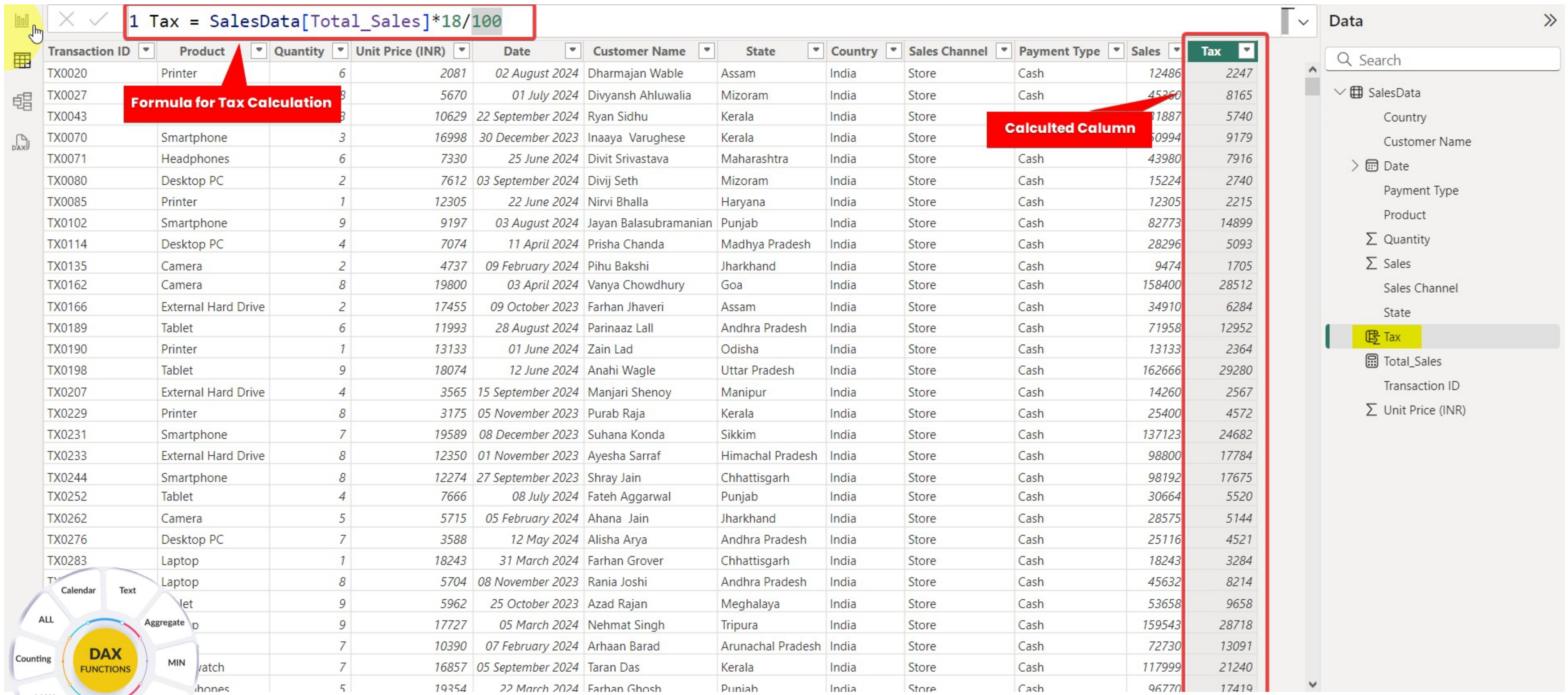
Calculated Column in Power BI



```
Tax = SalesData[Total_Sales]*18/100
```

Transaction ID	Product	Quantity	Unit Price (INR)	Date	Customer Name	State	Country	Sales Channel	Payment Type	Sales	Tax
TX0020	Printer	6	2081	02 August 2024	Dharmajan Wable	Assam	India	Store	Cash	12486	2247
TX0027	External Hard Drive	8	5670	01 July 2024	Divyansh Ahluwalia	Mizoram	India	Store	Cash	45360	8165
TX0043	Tablet	3	10629	22 September 2024	Ryan Sidhu	Kerala	India	Store	Cash	31887	5740
TX0070	Smartphone	3	16998	30 December 2023	Inaaya Varughese	Kerala	India	Store	Cash	50994	9179
TX0071	Headphones	6	7330	25 June 2024	Divit Srivastava	Maharashtra	India	Store	Cash	43980	7916
TX0080	Desktop PC	2	7612	03 September 2024	Divij Seth	Mizoram	India	Store	Cash	15224	2740
TX0085	Printer	1	12305	22 June 2024	Nirvi Bhalla	Haryana	India	Store	Cash	12305	2215
TX0102	Smartphone	9	9197	03 August 2024	Jayan Balasubramanian	Punjab	India	Store	Cash	82773	14899
TX0114	Desktop PC	4	7074	11 April 2024	Prisha Chanda	Madhya Pradesh	India	Store	Cash	28296	5093
TX0135	Camera	2	4737	09 February 2024	Pihu Bakshi	Jharkhand	India	Store	Cash	9474	1705
TX0162	Camera	8	19800	03 April 2024	Vanya Chowdhury	Goa	India	Store	Cash	158400	28512
TX0166	External Hard Drive	2	17455	09 October 2023	Farhan Jhaveri	Assam	India	Store	Cash	34910	6284
TX0189	Tablet	6	11993	28 August 2024	Parinaaz Lall	Andhra Pradesh	India	Store	Cash	71958	12952
TX0190	Printer	1	13133	01 June 2024	Zain Lad	Odisha	India	Store	Cash	13133	2364
TX0198	Tablet	9	18074	12 June 2024	Anahi Wagle	Uttar Pradesh	India	Store	Cash	162666	29280
TX0207	External Hard Drive	4	3565	15 September 2024	Manjari Shenoy	Manipur	India	Store	Cash	14260	2567
TX0229	Printer	8	3175	05 November 2023	Purab Raja	Kerala	India	Store	Cash	25400	4572
TX0231	Smartphone	7	19589	08 December 2023	Suhana Konda	Sikkim	India	Store	Cash	137123	24682
TX0233	External Hard Drive	8	12350	01 November 2023	Ayesha Sarraf	Himachal Pradesh	India	Store	Cash	98800	17784
TX0244	Smartphone	8	12274	27 September 2023	Shray Jain	Chhattisgarh	India	Store	Cash	98192	17675
TX0252	Tablet	4	7666	08 July 2024	Fateh Aggarwal	Punjab	India	Store	Cash	30664	5520
TX0262	Camera	5	5715	05 February 2024	Ahana Jain	Jharkhand	India	Store	Cash	28575	5144
TX0276	Desktop PC	7	3588	12 May 2024	Alisha Arya	Andhra Pradesh	India	Store	Cash	25116	4521
TX0283	Laptop	1	18243	31 March 2024	Farhan Grover	Chhattisgarh	India	Store	Cash	18243	3284
TX0290	Laptop	8	5704	08 November 2023	Rania Joshi	Andhra Pradesh	India	Store	Cash	45632	8214
TX0311	Tablet	9	5962	25 October 2023	Azad Rajan	Meghalaya	India	Store	Cash	53658	9658
TX0316	Laptop	9	17727	05 March 2024	Nehmat Singh	Tripura	India	Store	Cash	159543	28718
TX0317	Laptop	7	10390	07 February 2024	Arhaan Barad	Arunachal Pradesh	India	Store	Cash	72730	13091
TX0321	Smartwatch	7	16857	05 September 2024	Taran Das	Kerala	India	Store	Cash	117999	21240
TX0326	Headphones	5	19354	22 March 2024	Farhan Ghosh	Punjab	India	Store	Cash	96770	17419

Calculated Column in Power BI



```
1 Tax = SalesData[Total_Sales]*18/100
```

Formula for Tax Calculation

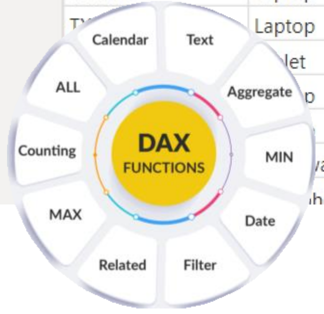
Calculted Column

Transaction ID	Product	Quantity	Unit Price (INR)	Date	Customer Name	State	Country	Sales Channel	Payment Type	Sales	Tax
TX0020	Printer	6	2081	02 August 2024	Dharmajan Wable	Assam	India	Store	Cash	12486	2247
TX0027	[illegible]	8	5670	01 July 2024	Divyansh Ahluwalia	Mizoram	India	Store	Cash	45360	8165
TX0043	[illegible]	3	10629	22 September 2024	Ryan Sidhu	Kerala	India	Store	[illegible]	31887	5740
TX0070	Smartphone	3	16998	30 December 2023	Inaaya Varughese	Kerala	India	Store	[illegible]	50994	9179
TX0071	Headphones	6	7330	25 June 2024	Divit Srivastava	Maharashtra	India	Store	Cash	43980	7916
TX0080	Desktop PC	2	7612	03 September 2024	Divij Seth	Mizoram	India	Store	Cash	15224	2740
TX0085	Printer	1	12305	22 June 2024	Nirvi Bhalla	Haryana	India	Store	Cash	12305	2215
TX0102	Smartphone	9	9197	03 August 2024	Jayan Balasubramanian	Punjab	India	Store	Cash	82773	14899
TX0114	Desktop PC	4	7074	11 April 2024	Prisha Chanda	Madhya Pradesh	India	Store	Cash	28296	5093
TX0135	Camera	2	4737	09 February 2024	Pihu Bakshi	Jharkhand	India	Store	Cash	9474	1705
TX0162	Camera	8	19800	03 April 2024	Vanya Chowdhury	Goa	India	Store	Cash	158400	28512
TX0166	External Hard Drive	2	17455	09 October 2023	Farhan Jhaveri	Assam	India	Store	Cash	34910	6284
TX0189	Tablet	6	11993	28 August 2024	Parinaaz Lall	Andhra Pradesh	India	Store	Cash	71958	12952
TX0190	Printer	1	13133	01 June 2024	Zain Lad	Odisha	India	Store	Cash	13133	2364
TX0198	Tablet	9	18074	12 June 2024	Anahi Wagle	Uttar Pradesh	India	Store	Cash	162666	29280
TX0207	External Hard Drive	4	3565	15 September 2024	Manjari Shenoy	Manipur	India	Store	Cash	14260	2567
TX0229	Printer	8	3175	05 November 2023	Purab Raja	Kerala	India	Store	Cash	25400	4572
TX0231	Smartphone	7	19589	08 December 2023	Suhana Konda	Sikkim	India	Store	Cash	137123	24682
TX0233	External Hard Drive	8	12350	01 November 2023	Ayesha Sarraf	Himachal Pradesh	India	Store	Cash	98800	17784
TX0244	Smartphone	8	12274	27 September 2023	Shray Jain	Chhattisgarh	India	Store	Cash	98192	17675
TX0252	Tablet	4	7666	08 July 2024	Fateh Aggarwal	Punjab	India	Store	Cash	30664	5520
TX0262	Camera	5	5715	05 February 2024	Ahana Jain	Jharkhand	India	Store	Cash	28575	5144
TX0276	Desktop PC	7	3588	12 May 2024	Alisha Arya	Andhra Pradesh	India	Store	Cash	25116	4521
TX0283	Laptop	1	18243	31 March 2024	Farhan Grover	Chhattisgarh	India	Store	Cash	18243	3284
[illegible]	Laptop	8	5704	08 November 2023	Rania Joshi	Andhra Pradesh	India	Store	Cash	45632	8214
[illegible]	[illegible]	9	5962	25 October 2023	Azad Rajan	Meghalaya	India	Store	Cash	53658	9658
[illegible]	[illegible]	9	17727	05 March 2024	Nehmat Singh	Tripura	India	Store	Cash	159543	28718
[illegible]	[illegible]	7	10390	07 February 2024	Arhaan Barad	Arunachal Pradesh	India	Store	Cash	72730	13091
[illegible]	[illegible]	7	16857	05 September 2024	Taran Das	Kerala	India	Store	Cash	117999	21240
[illegible]	[illegible]	5	19354	22 March 2024	Farhan Ghosh	Punjab	India	Store	Cash	96770	17419

Data

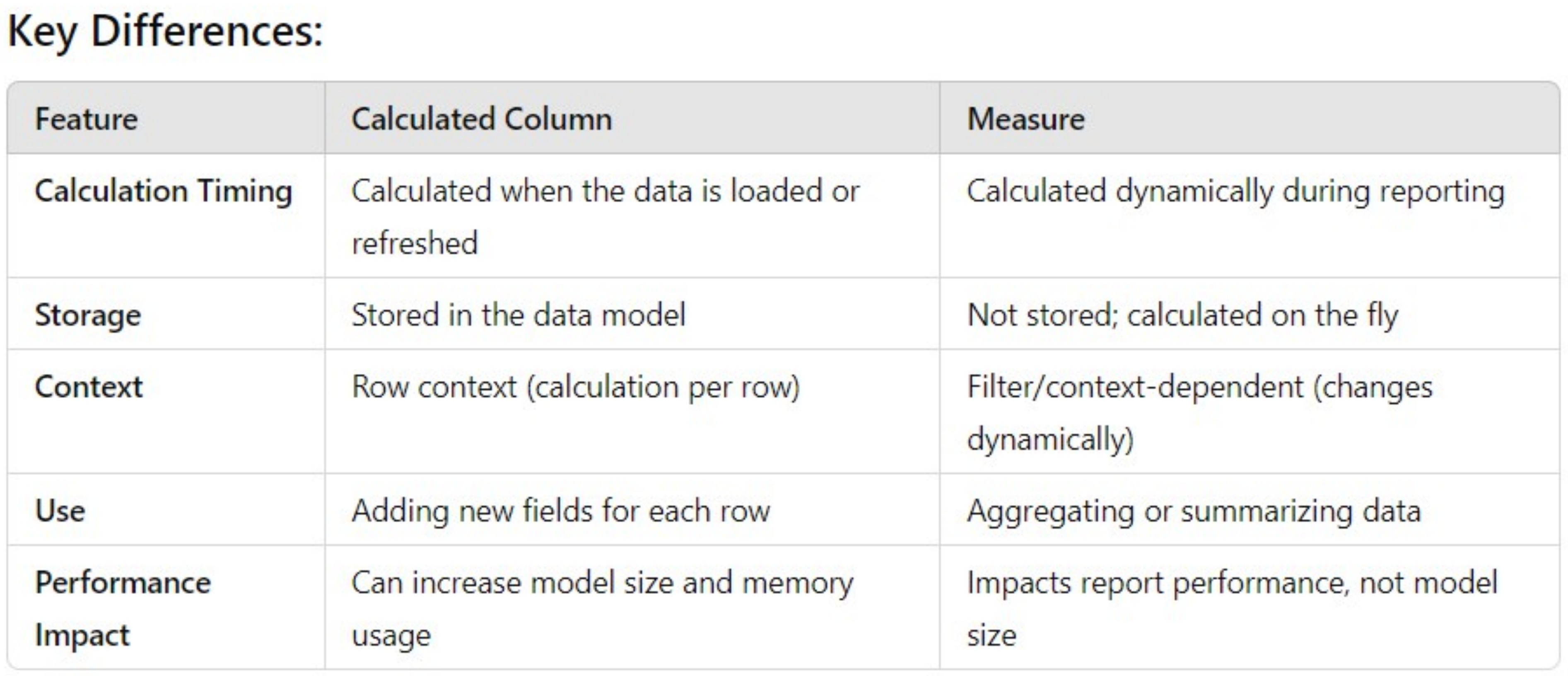
Search

- SalesData
  - Country
  - Customer Name
  - Date
  - Payment Type
  - Product
  - Σ Quantity
  - Σ Sales
  - Sales Channel
  - State
  - Tax
  - Total_Sales
  - Transaction ID
  - Σ Unit Price (INR)



DIFFERENCE BETWEEN CALCULATED COLUMN AND MEASURE IN POWER BI

In Power BI, both **calculated columns** and **measures** are used to perform calculations, but they serve different purposes and behave differently. Here's a comparison of the two:



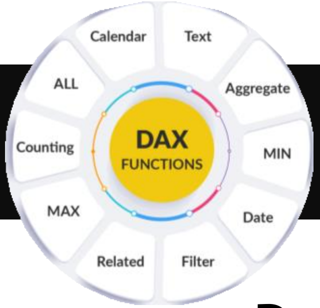
Key Differences:

Feature	Calculated Column	Measure
Calculation Timing	Calculated when the data is loaded or refreshed	Calculated dynamically during reporting
Storage	Stored in the data model	Not stored; calculated on the fly
Context	Row context (calculation per row)	Filter/context-dependent (changes dynamically)
Use	Adding new fields for each row	Aggregating or summarizing data
Performance Impact	Can increase model size and memory usage	Impacts report performance, not model size

SUMX FORMULA

DAX
FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

SUMX Formula in Power BI



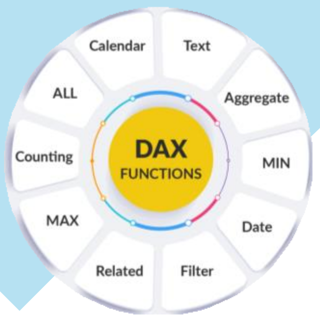
Definition: SUMX is an iterator function that performs row-by-row calculations and then sums the results.

Example

```
Total_Sales2 = SUMX(SalesData,SalesData[Quantity]*SalesData[Unit Price (INR)])
```

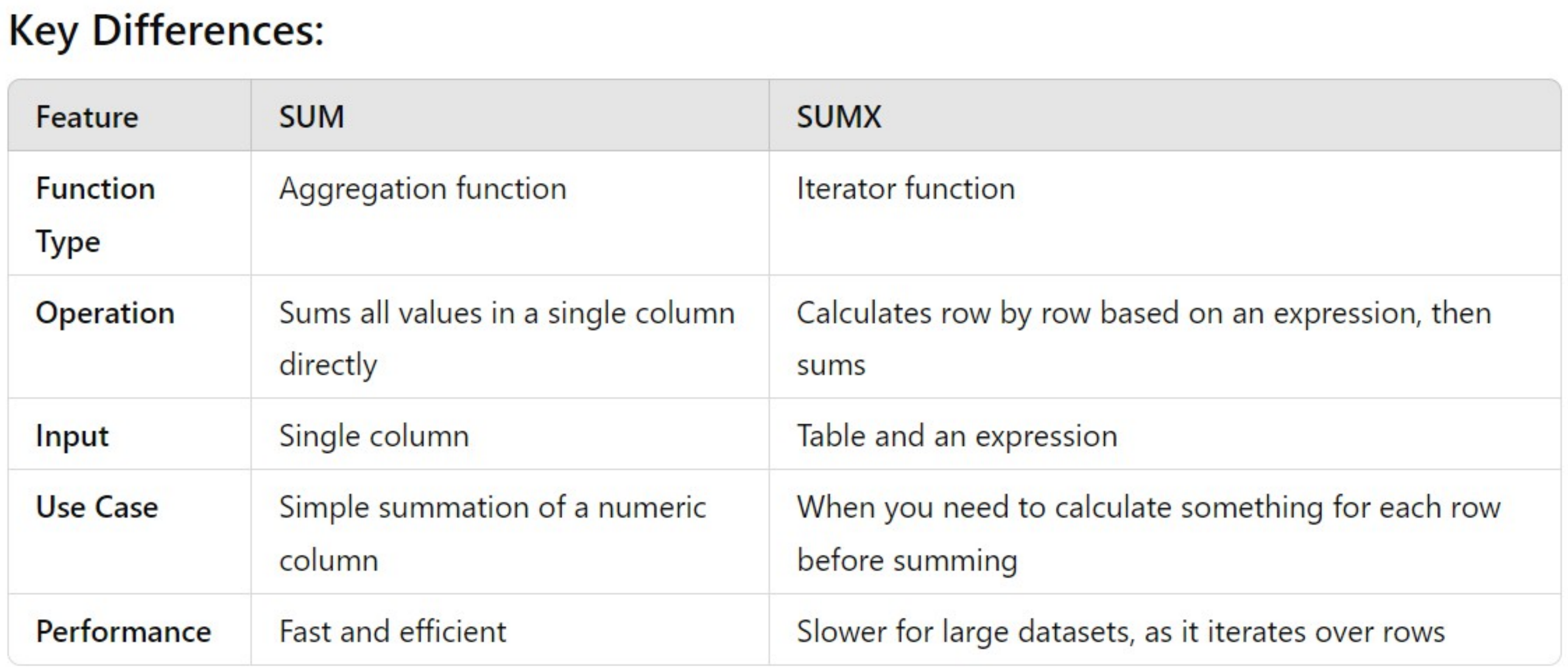
Use Case: Use SUMX when you need to perform a calculation for each row before summing, such as multiplying two columns together or applying conditional logic.

Performance: SUMX can be slower than SUM because it performs calculations on each row individually before summing. It's ideal for more complex scenarios that require row context.



Difference between SUM and SUMX Formula in Power BI

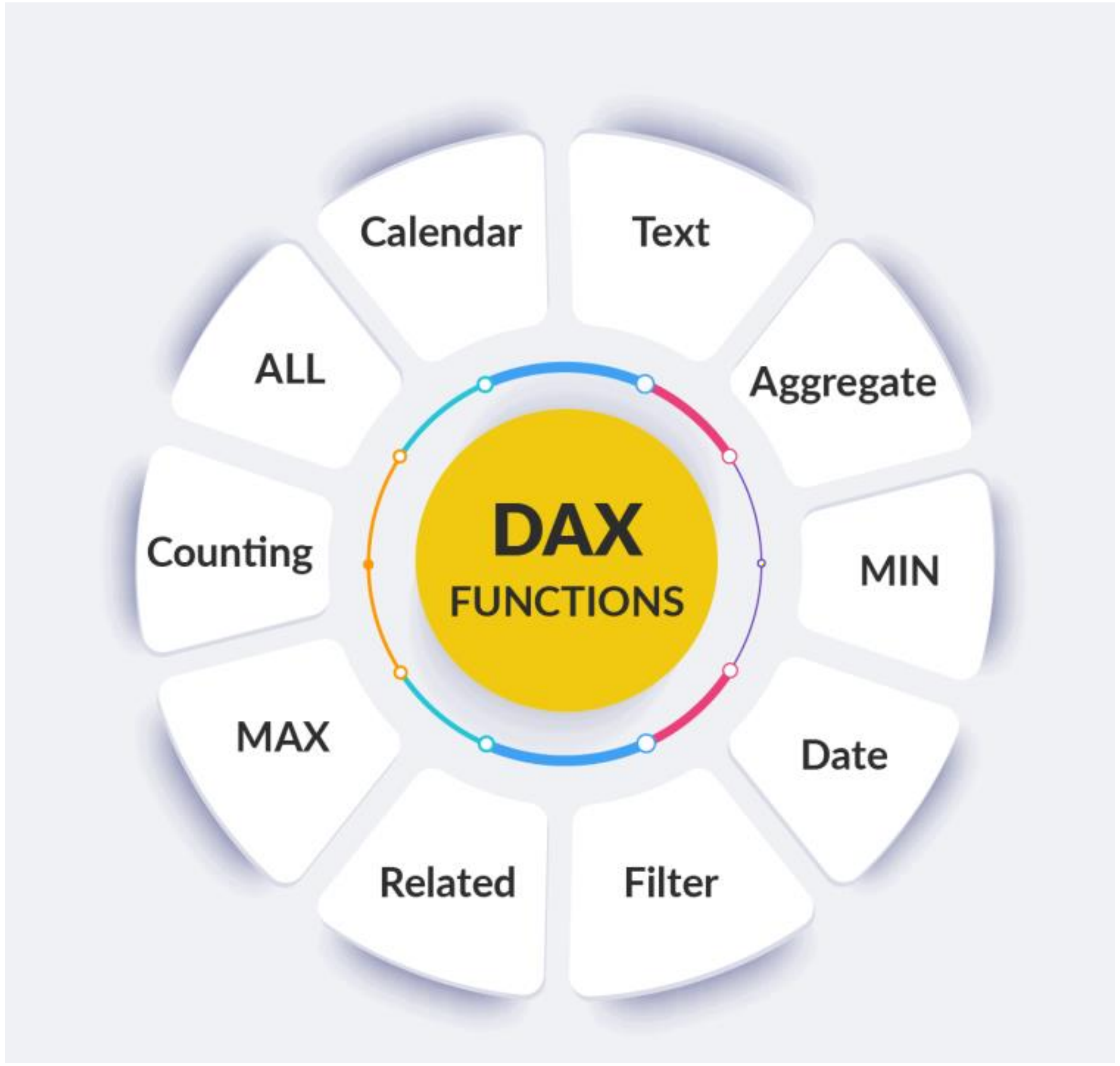
In Power BI, both SUM and SUMX are used to perform summation, but they work differently and are used in different scenarios. Here's a breakdown of the key differences:



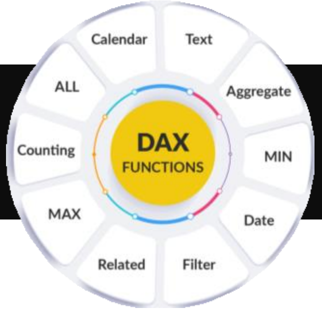
Key Differences:

Feature	SUM	SUMX
Function Type	Aggregation function	Iterator function
Operation	Sums all values in a single column directly	Calculates row by row based on an expression, then sums
Input	Single column	Table and an expression
Use Case	Simple summation of a numeric column	When you need to calculate something for each row before summing
Performance	Fast and efficient	Slower for large datasets, as it iterates over rows

COUNT FORMULA



Count Formula in Power BI



Definition: COUNT counts the number of non-blank values in a column.

Example

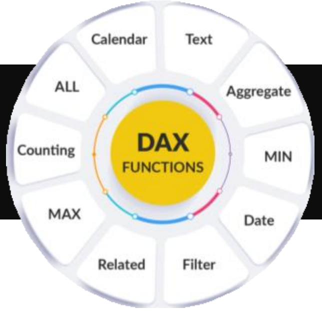
```
Number of Customers = COUNT('Table'[Customer ID])
```

Use Case: When you want to count the non-empty values in a column.

COUNTA FORMULA

DAX FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

COUNTA Formula in Power BI



Definition: COUNTA counts the number of non-blank values in a column, including text, numeric values, and logical values even counts Boolean values where only Count will not count Boolean data type

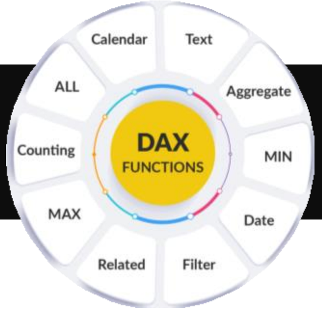
Example

```
Number of Reviews = COUNTA('Table'[Review Status])
```

Use Case: When you need to count all non-blank values, regardless of the data type (text, numbers, etc.).

COUNTBLANK FORMULA

COUNTBLANK Formula in Power BI



Definition: COUNTBLANK counts the number of blank (empty) values in a column.

Example

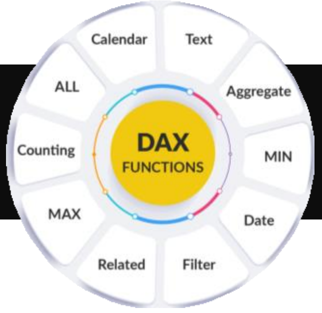
```
Blank Reveiws = COUNTBLANK('Table'[Review Points])
```

Use Case: When you want to count the number of blank or missing values in a column.

COUNTROWS FORMULA

Calendar
Text
ALL
Aggregate
DAX
FUNCTIONS
Counting
MIN
MAX
Date
Related
Filter

COUNTROWS Formula in Power BI



Definition: COUNTROWS counts the number of rows in a table.

Example

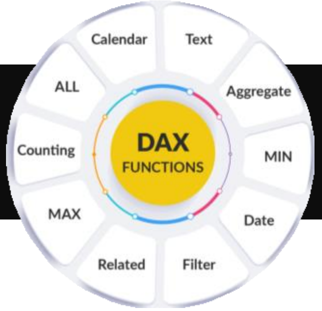
```
Total Records = COUNTROWS('Table')
```

Use Case: When you want to count the total number of rows in a table or in a filtered table.

DISTINCTCOUNT FORMULA

DAX
FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

DISTINCTCOUNT Formula



Definition: DISTINCTCOUNT counts the number of unique, non-blank values in a column.

Example

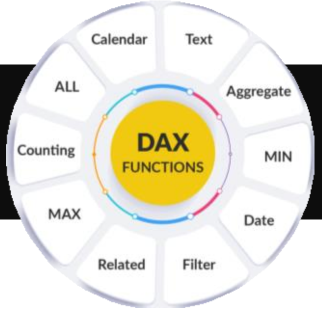
```
Unique States = DISTINCTCOUNT('Table'[State])
```

Use Case: When you need to count the distinct (unique) values in a column, excluding blanks.

COUNTX FORMULA

DAX FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

COUNTX Formula



Definition: COUNTX is an iterator function that counts non-blank results of an expression evaluated row by row over a table.

Example

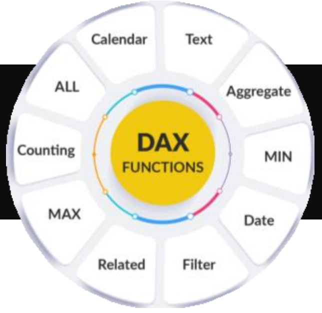
```
Count Reviews >= 5 = COUNTX('Table', IF('Table'[Review Points] >= 5, 1, BLANK()))
```

Use Case: When you need to count based on an expression that is evaluated for each row in a table.

COUNTAX FORMULA

DAX FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

COUNTAX Formula



Definition: COUNTAX is the iterator version of COUNTA. It evaluates an expression for each row and counts the number of non-blank results.

Example

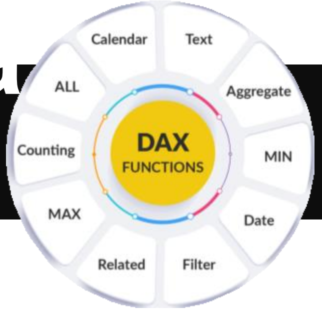
```
Count True = COUNTAX(FILTER('Table','Table'[Review Status]=true),'Table'[Review
Status])
```

Use Case: When you need to count based on an expression that is evaluated for each row in a table even there is Binary data type.

COUNTROWS with FILTER FORMULA

DAX
FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

COUNTROWS with FILTER Formula

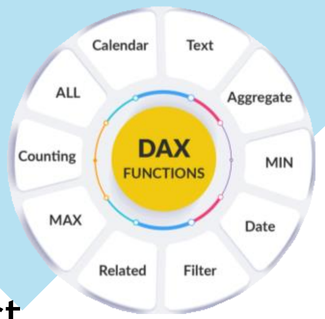


Definition: COUNTROWS can be used with the FILTER function to count rows that meet specific criteria.

Example

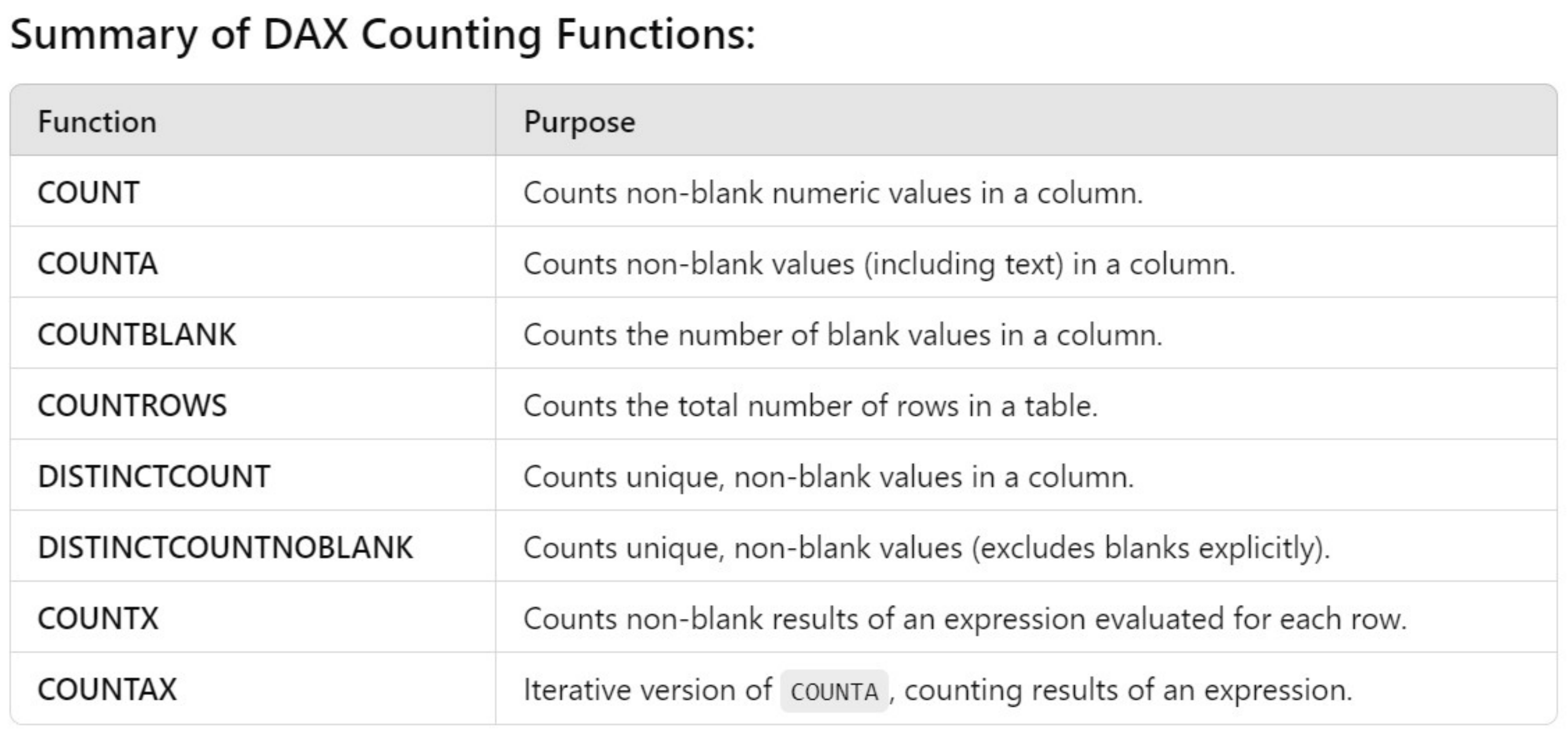
```
Maharashtra Count = COUNTROWS(FILTER('Table','Table'[State]="Maharashtra"))
```

Use Case: When you want to count rows based on a condition or set of conditions.



Summary of DAX Counting Functions:

These functions provide flexibility depending on whether you're counting all rows, distinct values, non-blank values, or values based on expressions and conditions.



Summary of DAX Counting Functions:

Function	Purpose
COUNT	Counts non-blank numeric values in a column.
COUNTA	Counts non-blank values (including text) in a column.
COUNTBLANK	Counts the number of blank values in a column.
COUNTROWS	Counts the total number of rows in a table.
DISTINCTCOUNT	Counts unique, non-blank values in a column.
DISTINCTCOUNTNOBLANK	Counts unique, non-blank values (excludes blanks explicitly).
COUNTX	Counts non-blank results of an expression evaluated for each row.
COUNTAX	Iterative version of `COUNTA`, counting results of an expression.

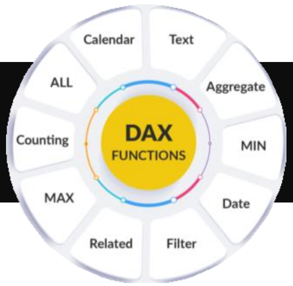
Date Formulas

Calendar
Text
ALL
Aggregate
DAX
FUNCTIONS
Counting
MIN
MAX
Date
Related
Filter

Extract Day/ Month/Year

DAX
FUNCTIONS
Calendar
Text
Aggregate
MIN
Date
Filter
Related
MAX
Counting
ALL

Extract Day/Month/Year



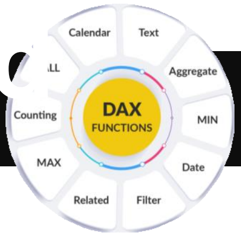
DAY(<datetime>) : Extracts the day from a date value.

MONTH(<datetime>) : Extracts the month from a date value.

YEAR(<datetime>) : Extracts the year from a date value.

Satish Dhawale

Extract Hour/Minute/Secon

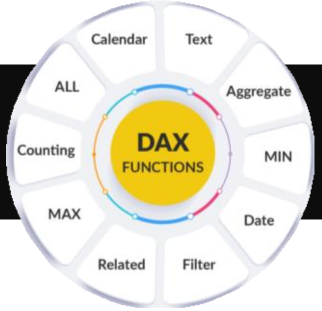


Hour(<datetime>) : Extracts the Hour from a date and time value.

Minute(<datetime>) : Extracts the Minute from a date and time value.

Second(<datetime>) : Extracts the Second from a date and time value.

Extract Hour/Minute/Second



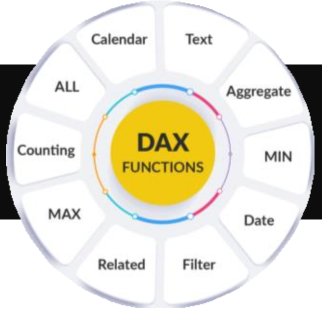
Today() : Show Current date

Now() – Show current date and Time

Weekday() – Show Weekday in numbers between 1 to 7

Weeknum() – Show Week number in Month/Year

Datediff Formula



Definition: Returns the difference between two dates in the specified interval (days, months, years, etc.).

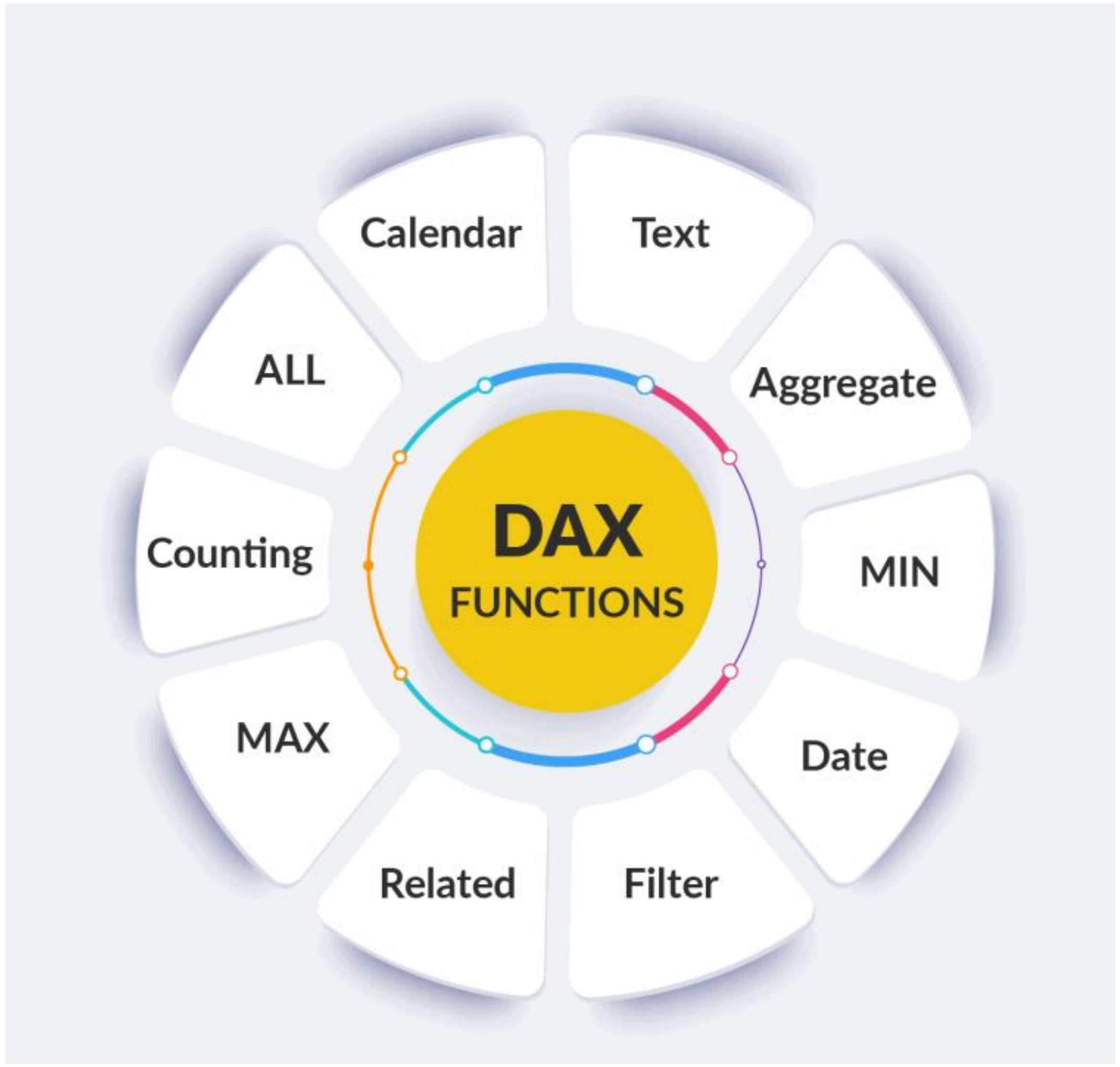
Syntax

```
DATEDIFF(<start_date>, <end_date>, <interval>)
```

Example

```
DATEDIFF(Sales[OrderDate], Sales[ShipDate], DAY)
```

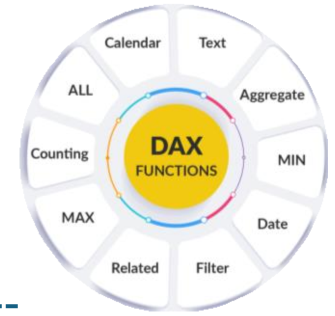
Create Custom Calendar



1 . Create Custom Calendar in Power Query

Formula

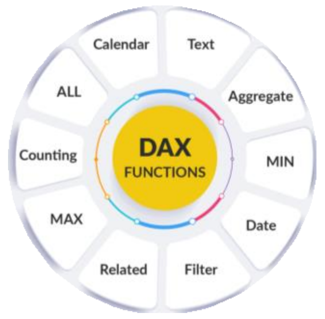
```
= List.Dates(#date(2023,1,1),731,#duration(1,0,0,0))
```



Creating a **custom calendar** using **Power Query** in Power BI is a common and essential task when dealing with date-related data, especially for performing time-based analysis like year-over-year (YoY), month-to-date (MTD), and quarter-to-date (QTD) comparisons. Having a custom date table ensures that you have control over the format, date ranges, and any specific time-based logic needed for your reports.

2. Create Custom Calendar in Power Query

M Code



```
let
    // Define start and end dates for the calendar
    StartDate = #date(2020, 1, 1),// You can change this to your desired start date
    EndDate = Date.From(DateTime.LocalNow()),  // Automatically takes today's date as the end date
        // Generate a list of dates between the start and end dates
    DateList = List.Dates(StartDate, Duration.Days(EndDate - StartDate) + 1, #duration(1, 0, 0, 0)),
    // Convert the list into a table
    DateTable = Table.FromList(DateList, Splitter.SplitByNothing(), {"Date"}, null, ExtraValues.Error),
    // Add Year, Month, and Day columns for further analysis
    AddYear = Table.AddColumn(DateTable, "Year", each Date.Year([Date]),
    Int64.Type), AddMonth = Table.AddColumn(AddYear, "Month", each
    Date.Month([Date]), Int64.Type), AddDay = Table.AddColumn(AddMonth, "Day", each
    Date.Day([Date]), Int64.Type),
    AddMonthName = Table.AddColumn(AddDay, "Month Name", each Date.ToText([Date], "MMMM"), type text),
    AddQuarter = Table.AddColumn(AddMonthName, "Quarter", each Date.QuarterOfYear([Date]),
    Int64.Type), AddYearMonth = Table.AddColumn(AddQuarter, "Year-Month", each Date.ToText([Date],
    "yyyy-MM"), type

text),
    AddWeekOfYear = Table.AddColumn(AddYearMonth, "Week of Year", each Date.WeekOfYear([Date]),
Int64.Type),
    AddDayOfWeek = Table.AddColumn(AddWeekOfYear, "Day of Week", each Date.ToText([Date],
"dddd"), type text),
    AddIsWeekend = Table.AddColumn(AddDayOfWeek, "Is Weekend", each if Date.DayOfWeek([Date],
Day.Sunday) > 4 then "Yes" else "No", type text)
in
```

AddIsWeekend

3 . Calendar DAX Formula

Formula

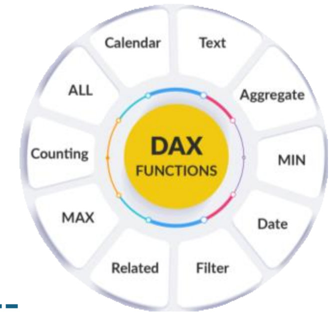
```
= CALENDAR(MIN(SalesData[Date]),MAX(SalesData[Date]))
```

The CALENDAR DAX function in Power BI is used to generate a continuous range of dates between a specified start and end date. It's particularly useful when building a date table, which is essential for time-based calculations such as year-over-year analysis, month-to-date, quarter-to-date, etc.

4 . CalendarAuto DAX Formula

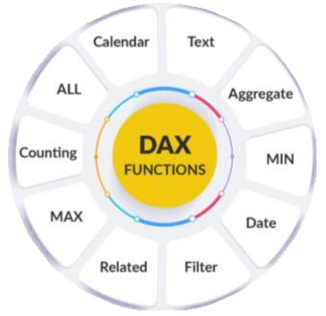
Formula

```
= CALENDARAUTO()
```



The CALENDARAUTO function in Power BI is a powerful tool for creating a date table that automatically detects the date range from all date columns in your data model. This means it scans your data and generates a date range based on the minimum and maximum dates found in the model, which is especially helpful for dynamic date tables without specifying start or end dates manually.

MTD QTD AND YTD DAX FORMULA

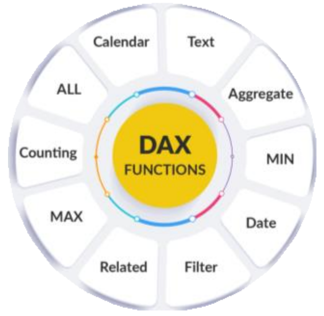


Formula

```
Sale  MT  = TOTALMTD(SUM(Sales[SalesAmount  DateTable[Date
 s     D                     ]),                    ])
Sale  QT  = TOTALQTD(SUM(Sales[SalesAmount  DateTable[Date
 s     D                     ]),                    ])
Sale  YT  = TOTALYTD(SUM(Sales[SalesAmount  DateTable[Date
 s     D                     ]),                    ])
```

MTD (Month-to-Date), QTD (Quarter-to-Date), and YTD (Year-to-Date) are commonly used DAX functions in Power BI for performing time-based aggregations. They are particularly helpful for tracking progress over the current month, quarter, or year, making it easy to see cumulative totals up to the present date. These functions rely on having a properly structured date table, ideally linked to your data model.

Networkdays DAX Formula



Formula

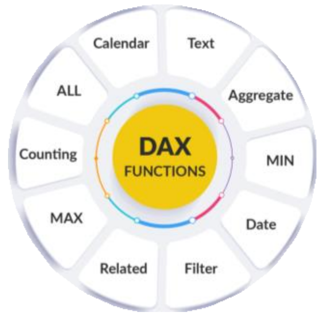
```
Network Days = NETWORKDAYS("01-01-2023","31-01-2023",1,Holidays)
```

In Power BI, Network Days refers to the count of working days (typically Monday through Friday) between two specified dates, excluding weekends and optionally excluding holidays.

Networkdays DAX Formula

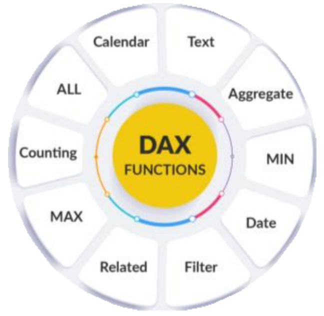
```
Network Days = NETWORKDAYS("01-01-2023","31-01-2023",1,Holidays)
```

1 or omitted: Saturday, Sunday
2: Sunday, Monday
3: Monday, Tuesday
4: Tuesday, Wednesday
5: Wednesday, Thursday
6: Thursday, Friday
7: Friday, Saturday
11: Sunday only
12: Monday only
13: Tuesday only
14: Wednesday only
15: Thursday only
16: Friday only
17: Saturday only



DATESINPERIOD DAX FORMULA

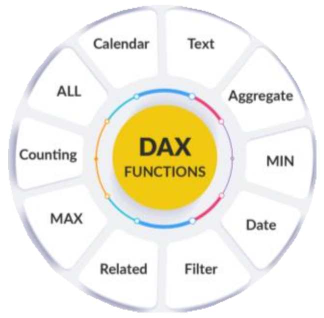
Formula



```
DATESINPERIOD(<dates>, <start_date>, <number_of_intervals>, <interval>)
```

The DATESINPERIOD function in DAX returns a single-column table of dates within a specified period, based on a start date and a defined interval. It’s useful when you need to create calculations over dynamic date ranges (e.g., last 7 days, previous month, etc.).

DATESBETWEEN DAX FORMULA

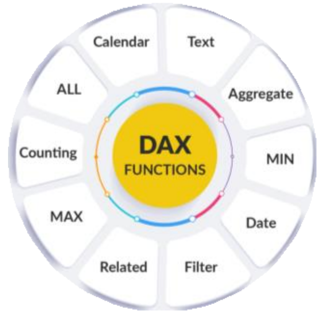


Formula

```
DATESBETWEEN(<dates>, <start_date>, <end_date>)
```

The DATESBETWEEN function in DAX returns a table with dates within a specified start and end date range. This function is useful when you want to filter data to specific date boundaries.

SAME PERIOD LAST YEAR DAX FORMULA



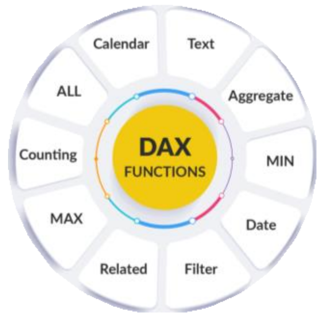
Formula

```
SAMEPERIODLASTYEAR(<dates>)
```

The SAMEPERIODLASTYEAR function in DAX is used to return a table with the dates for the same period in the previous year. This function is often used in time intelligence calculations to compare current performance to the previous year.

CALCULATE DAX FORMULA

Formula



```
SAMEPERIODLASTYEAR(<dates>)
```

The SAMEPERIODLASTYEAR function in DAX is used to return a table with the dates for the same period in the previous year. This function is often used in time intelligence calculations to compare current performance to the previous year.

TEXT FORMULAS

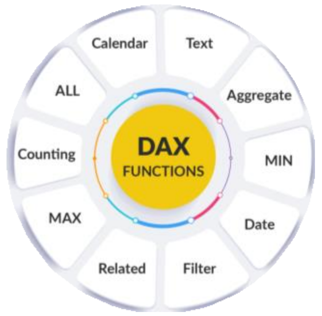
In Power BI, there are several DAX functions designed for working with text data. These functions allow you to manipulate and format text values in various ways. Here are some commonly used text DAX formulas in Power BI:

LEFT, RIGHT, MID

```
LEFT("Power BI", 5)  // Result: "Power"

RIGHT("Power BI", 2)  // Result: "BI"

MID("Power BI", 7, 2)  // Result: "BI"
```



TEXT FORMULAS

In Power BI, there are several DAX functions designed for working with text data. These functions allow you to manipulate and format text values in various ways. Here are some commonly used text DAX formulas in Power BI:

UPPER and LOWER

```
UPPER("Power BI")  // Result: "POWER BI"
LOWER("Power BI")  // Result: "power bi"
```

LEN: Returns the length (number of characters) of a text string.