POWERBI COMPLETE DAX GUIDE

Data Analysis Expressions (DAX) is a library of functions and operators that can be combined to build formulas and expressions in Power BI, Analysis Services, and Power Pivot in Excel data models.

Aggregation functions overview

AVERAGE

```
AVERAGE(<column>)
If the column contains text, no aggregation can be performed, and
the functions returns blanks.
If the column contains logical values or empty cells, those values
are ignored.
Cells with the value zero are included.

This function is not supported for use in DirectQuery mode when us
ed in calculated columns or row-level security (RLS) rules.
```

AVERAGEA

```
AVERAGEA(<column>)



also handles non-numeric data types according to the following rul
es:

Values that evaluates to TRUE count as 1.
Values that evaluate to FALSE count as 0 (zero).
Values that contain non-numeric text count as 0 (zero).
Empty text ("") counts as 0 (zero).



Transaction ID  Amount  Result
0000123          1       Counts as 1
0000124         20           Counts as 20
0000125         n/a          Counts as 0
0000126                      Counts as 0
0000126         TRUE         Counts as 1
```

AVERAGEX

AVERAGEX follows the same rules as AVERAGE. You cannot include non-numeric or null cells. Both the table and expression arguments are required.

COUNT

The COUNT function counts rows that contain the following kinds of values:

Numbers
Dates
Strings

When the function finds no rows to count, it returns a blank.
Blank values are skipped. TRUE/FALSE values are not supported.
If you want to evaluate a column of TRUE/FALSE values, use the COUNTA function.

COUNTA

Unlike COUNT, COUNTA supports Boolean data type.

COUNTX

The COUNTX function counts only values, dates, or strings. If the function finds no rows to count, it returns a blank.
If you want to count logical values, use the COUNTAX function.

```
= COUNTX(FILTER(Product,RELATED(ProductSubcategory[EnglishProductSubcategoryName])="Caps"), Product[ListPrice])
```

COUNTAX

The COUNTAX function counts non-blank results when evaluating the result of an expression over a table.
COUNTAX(<table>,<expression>)

= COUNTAX(FILTER('Reseller',[Status]="Active"),[Phone])

COUNTBLANK

Counts the number of blank cells in a column.
COUNTBLANK(<column>)

COUNTROWS

This function can be used to count the number of rows in a base ta ble
COUNTROWS([<table>])

DISTINCTCOUNT

Counts the number of distinct values in a column.
DISTINCTCOUNT(<column>)
DISTINCTCOUNT function ALSO counts the BLANK valu

DISTINCTCOUNTNOBLANK

Counts the number of distinct values in a column.
does not count the BLANK value.
= DISTINCTCOUNT(ResellerSales_USD[SalesOrderNumber])

MAX

Returns the largest value in a column, or between two scalar expressions.
MAX(<column>)
MAX(<expression1>, <expression2>)

When comparing two expressions, blank is treated as 0 when comparing.
That is, Max(1, Blank()) returns 1, and Max(-1, Blank()) returns 0.
If both arguments are blank, MAX returns a blank.
If either expression returns a value which is not allowed, MAX returns an error.

TRUE/FALSE values are not supported. If you want to evaluate a column of TRUE/FALSE values, use the MAXA function.

= Max([TotalSales], [TotalPurchases])

MAXA

Returns the largest value in a column.

MAXA(<column>)
looks for the largest value among the following types of values:
Numbers
Dates

Logical values, such as TRUE and FALSE. Rows that evaluate to TRUE count as 1;
rows that evaluate to FALSE count as 0 (zero).

Empty cells are ignored. If the column contains no values that can be used, MAXA returns 0 (zero).
If you want to compare text values, use the MAX function.

= MAXA([ResellerMargin])
= MAXA([TransactionDate])

MAXX

Returns the highest value that results from evaluating an expressi
on for each row of a table.
MAXX(<table>,<expression>,[<variant>])

variant: (Optional) If TRUE, and if there are variant or mixed val
ue types,
the highest value based on ORDER BY DESC is returned.

Of the values to evaluate, only the following are counted:
Numbers
Texts
Dates

Blank values are skipped. TRUE/FALSE values are not supported.

If the expression has variant or mixed value types such as text an
d number,
then by default MAXX considers only numbers. If <variant> = TRUE,
the maximum value is returned.

= MAXX(FILTER(InternetSales,[SalesTerritoryCode]="5"), InternetSal
es[TaxAmt]+ InternetSales[Freight])

MIN

Returns the smallest value in a column, or between two scalar expr
essions.
MIN(<column>)
MIN(<expression1>, <expression2>)

The following types of values in the columns are counted:
Numbers
Texts
Dates
Blanks

When comparing expressions, blank is treated as 0 when comparing.
That is, Min(1,Blank()) returns 0, and Min(-1, Blank()) returns
-1.
If both arguments are blank, MIN returns a blank.
If either expression returns a value which is not allowed, MIN ret
urns an error.

TRUE/FALSE values are not supported. If you want to evaluate a col
umn of TRUE/FALSE values, use the MINA function.

MINA

MINA(<column>)
Returns the smallest value in a column.
If the column contains no values, MINA returns 0 (zero).
Rows in the column that evaluates to logical values,
such as TRUE and FALSE are treated as 1 if TRUE and 0 (zero) if FA
LSE.Empty cells are ignored.

MINX

Returns the lowest value that results from evaluating an expressio
n for each row of a table.
MINX(<table>, < expression>,[<variant>])

variant: (Optional) If TRUE, and if there are variant or mi
xed value types,
the lowest value based on ORDER BY ASC is returned.

Blank values are skipped. TRUE/FALSE values are not supported.

If the expression has variant or mixed value types such as text an
d number,
then by default MINX considers only numbers. If <variant> = TRUE,
the minimum value is returned.

= MINX(FILTER(InternetSales, [SalesTerritoryKey] = 5),[Freight])

= MINX(FILTER(InternetSales, InternetSales[SalesTerritoryKey] =
5), InternetSales[Freight] + InternetSales[TaxAmt])

PRODUCT

Returns the product of the numbers in a column.
PRODUCT(<column>)
Return value: A decimal number.

Only the numbers in the column are counted. Blanks, logical value
s, and text are ignored. For example,
PRODUCT(Table[Column]) is equivalent to PRODUCTX(Table, Table[C
olumn]).

PRODUCTX

Returns the product of an expression evaluated for each row in a t
able.
PRODUCTX(<table>, <expression>)
Only the numbers in the column are counted. Blanks, logical value
s, and text are ignored.

= [PresentValue] * PRODUCTX(AnnuityPeriods, 1+[FixedInterestRate]
)

SUM

Adds all the numbers in a column.
SUM(<column>)

SUMX

Returns the sum of an expression evaluated for each row in a tabl
e.
SUMX(<table>, <expression>)

Only the numbers in the column are counted. Blanks, logical value
s, and text are ignored.
= SUMX(FILTER(InternetSales, InternetSales[SalesTerritoryID]=5),[F
reight])

In [ ]:
```
1
```

Date and time functions overview

These functions help you create calculations based on dates and time. Many of the functions in DAX are similar to the Excel date and time functions. However, DAX functions use a datetime data type, and can take values from a column as an argument.

CALENDAR

Returns a table with a single column named "Date" that contains a
contiguous set of dates.
CALENDAR(<start_date>, <end_date>)

An error is returned if start_date is greater than end_date.
= CALENDAR (DATE (2015, 1, 1), DATE (2021, 12, 31))
= CALENDAR (MINX (Sales, [Date]), MAXX (Forecast, [Date]))