

## POWER BI DAX Functions

### CALCULATE

CALCULATE allows you to modify the filter context within which a calculation is evaluated.

**Formula:** CALCULATE(<expression>, <filter1>, <filter2>, ...)

**Example:** If you want to calculate total sales based on our net sales data, we can use the CALCULATE function. = CALCULATE(SUM('Sales'[NetSales]), 'Sales'[Year] = 2023)

### SUM

SUM adds up all the values in a specified column. It's a basic aggregate function used to get the total of numeric data.

**Formula:** =SUM(<column>)

**Example:** This calculates the total net sales by summing all values in the Sales[Net Sales] column. = SUM(Sales[Net Sales])

### SUMX

SUMX returns the sum of an expression evaluated for each row in a table. Unlike SUM, which operates on a single column, SUMX allows for row-by-row calculations.

**Formula:** =SUMX(<table>, <expression>)

**Example:** Here, for each row in the Sales table, it multiplies Sales[Quantity] by Sales[Profit per Unit] and then sums up those results. = SUMX(Sales, Sales[Quantity] \* Sales[Profit per Unit])

### AVERAGE

AVERAGE calculates the arithmetic mean of all the numbers in a specified column. It's useful for determining the central value of a set of numbers.

**Formula:** =AVERAGE(<column>) =AVERAGE(Sales[Net Sales])

**Example:** Here, This calculates the average of all values in the Sales[Net Sales] column.

### IF

IF checks a condition and returns one value if the condition is TRUE and another value if the condition is FALSE. It's used for conditional logic.

**Formula:** =IF(<logical\_test>, <value\_if\_true>, <value\_if\_false>)

**Example:** This checks if Sales[Net Sales] is greater than 10,000. If true, it returns "Yes"; otherwise, it returns "No". = IF(Sales[Net Sales] > 10000, "Yes", "No")

### FILTER

FILTER returns a table that represents a subset of another table or expression, based on a specified condition. It's used to create more specific views of data.

**Formula:** =FILTER(<table>, <filter>)

**Example:** It returns a table of sales data where the Region column equals "North". = FILTER(Sales, Sales[Region] = "North")

### ALL

ALL removes all filters from a specified table or column. It's often used in combination with other functions to provide a baseline or total value that isn't affected by the current filters.

**Formula:** =ALL(<table\_or\_column>)

**Example:** This calculates the total net sales without considering any existing filters on the Sales table. = CALCULATE(SUM(Sales[Net Sales]), ALL(Sales))

### **ALLEXCEPT**

ALLEXCEPT removes all context filters in a table except filters that have been applied to specified columns. This allows for selective removal of filters.

**Formula:** =ALLEXCEPT(<table>, <column1>, <column2>, ...)

**Example:** This keeps all filters on the Sales[Product Category] column but removes other filters, and then calculates the total net sales. = CALCULATE(SUM(Sales[Net Sales]), ALLEXCEPT(Sales, Sales[Product Category]))

### **RELATED**

RELATED returns a related value from another table. It's used to bring in data from a related table in a data model.

**Formula:** =RELATED(<column>)

**Example:** = RELATED(Product[Product Name]) Assuming there is a relationship between the Sales and Product tables, this returns the product name for each sale.

### **DISTINCT**

DISTINCT returns a one-column table that contains the unique values from the specified column. It's useful for identifying or counting distinct items in a column.

**Formula:** =DISTINCT(<column>)

**Example:** This returns a table of unique values from the Sales[Product Category] column. = DISTINCT(Sales[Product Category])