

Data Analysis Expressions

(https://shorturl.at/lewBK)







# DAX Basics

#### Measure:

**Measures** typically utilize **aggregate functions** to summarize data based on the current report context.

Example: Total Sales: Total Sales = SUM(Sales[SalesAmount])

**Table Name** 

Column name

#### Calculated Column:

Calculated Columns typically use arithmetic functions to compute values based on the data in individual rows.

Example: Total Price: Total Price = Products[UnitPrice] \* Products[Quantity]

**Table Name** 





## **Aggregate Functions:**

SUM: Adds up all the values in a column.

**AVERAGE**: Calculates the average of the values in a column.

**COUNT:** Counts the number of rows in a column that contain numbers

MIN: Returns the smallest value in a column.

MAX: Returns the largest value in a column.

# **Aggregate Functions**

## SUM

What is the total sales amount for the entire year?

Total Sales = SUM(Sales[SalesAmount])

## AVERAGE

What is the average sales amount per transaction?

Average Sales = AVERAGE(Sales[SalesAmount])

### COUNT

How many orders have been placed?

Total Orders = COUNT(Sales[OrderID])



# **Aggregate Functions**

## MIN

What is the lowest price of any product?

Lowest Price = MIN(Products[UnitPrice])

## MAX

What is the highest price of any product?

Highest Price = MAX(Products[UnitPrice])

## Logical Functions:

IF: Evaluates a condition and returns different values based on whether the condition is true or false.

**SWITCH**: Evaluates an expression against a list of values and returns the corresponding result for the first match.

**AND:** Returns TRUE if all arguments are TRUE.

**OR**: Returns TRUE if any argument is TRUE.

**NOT**: Returns the opposite of the logical value.

# Logical Functions

### • IF

Check if a sales amount is above a threshold.

#### SWITCH

Let's say you have a table of sales data, and you want to categorize sales amounts into different ranges: "Low", "Medium", and "High".

```
SalesCategory = SWITCH(TRUE(),

Sales[TotalSales] < 500, "Low",

Sales[TotalSales] < 2000, "Medium",

"High")

Column name
```



# Logical Functions

#### AND

Check if a product is both in stock and on sale

IsAvailable = IF(AND(Products[Stock] > 0, Products[OnSale] = TRUE),
"Available", "Not Available")

#### • OR

Determine if a customer qualifies for a special offer.

SpecialOffer = IF(OR(Customers[PurchaseCount] > 10,
Customers[LoyaltyStatus] = "Gold"), "Eligible", "Not Eligible")

### NOT

Check if a product is not discontinued.

IsNotDiscontinued = IF(NOT(Products[Discontinued]), "Active",
"Discontinued")



## **Date and Time Functions:**

**TODAY:** Returns the current date.

NOW: Returns the current date and time.

YEAR: Extracts the year from a date.

**MONTH**: Extracts the month from a date.

**DAY:** Extracts the day from a date.

**DATEDIFF**: Calculates the difference between two dates in a specified interval (e.g., DAY, MONTH, YEAR).

## Date and Time Functions

### · TODAY()

What DAX expression would you use to get the current date for your report?

CurrentDate = TODAY()

### • **NOW()**

How would you capture the exact date and time of each entry using DAX?

EntryTimestamp = NOW()

### YEAR(date)

How would you extract the year from a sales date column called Sales[OrderDate]?

SalesYear = YEAR(Sales[OrderDate])



## **Date and Time Functions**

## MONTH(date)

What DAX expression would you use to get the month from the Sales[OrderDate]?

SalesMonth = MONTH(Sales[OrderDate])

## • DAY(date)

How would you extract the day from the Sales[OrderDate]?

SalesDay = DAY(Sales[OrderDate])

#### DATEDIFF

How would you calculate the number of days between the Tickets[CreatedDate] and Tickets[ResolvedDate]?

ResolutionTime = DATEDIFF(Tickets[CreatedDate], Tickets[ResolvedDate], DAY)



## **Text Functions:**

**CONCATENATE:** Joins two or more text strings into one.

LEFT: Returns the leftmost characters from a text string.

RIGHT: Returns the rightmost characters from a text string.

LEN: Returns the number of characters in a text string.

SEARCH: Finds the position of a substring within a string.

**UPPER**: Converts text to uppercase

LOWER: Converts text to lowercase.

## **Text Functions**

#### CONCATENATE

If you have Customer[FirstName] and Customer[LastName], how would you create a full name?

FullName = CONCATENATE(Customer[FirstName], Customer[LastName]).

#### LEFT

How would you get the first three characters from Products[ProductCode]?

ProductPrefix = LEFT(Products[ProductCode], 3).

#### RIGHT

How would you get the last two digits from Employees[EmployeeID]?

LastTwoDigits = RIGHT(Employees[EmployeeID], 2).



# **Text Functions**

#### • LEN

How would you find the length of Products[Description]?

DescriptionLength = LEN(Products[Description])

#### UPPER

How would you convert Products[ProductName] to uppercase?

UpperCaseProductName = UPPER(Products[ProductName])

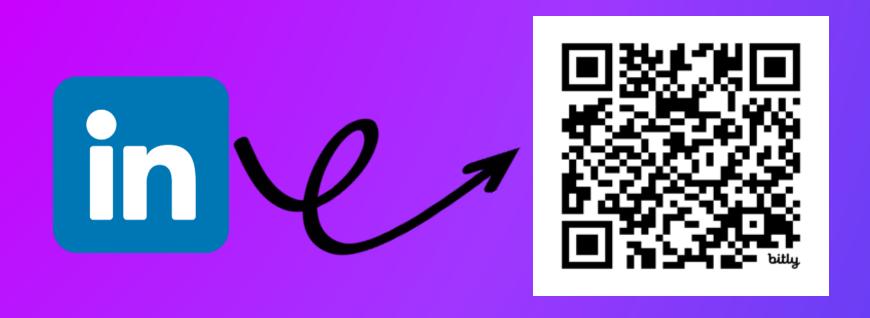
#### LOWER

How would you convert Customers[Email] to lowercase?

LowerCaseEmail = LOWER(Customers[Email])



follow Pranav Borge on LinkedIn for valuable insights on topics like data science, machine learning and Data Visualization.



(https://shorturl.at/lewBK)