# PHÂN TÍCH DỮ LIỆU VỚI DAX TRONG POWER BI

# PHÂN TÍCH DỮ LIỆU VỚI DAX TRONG POWER BI

01. Tổng quan về DAX

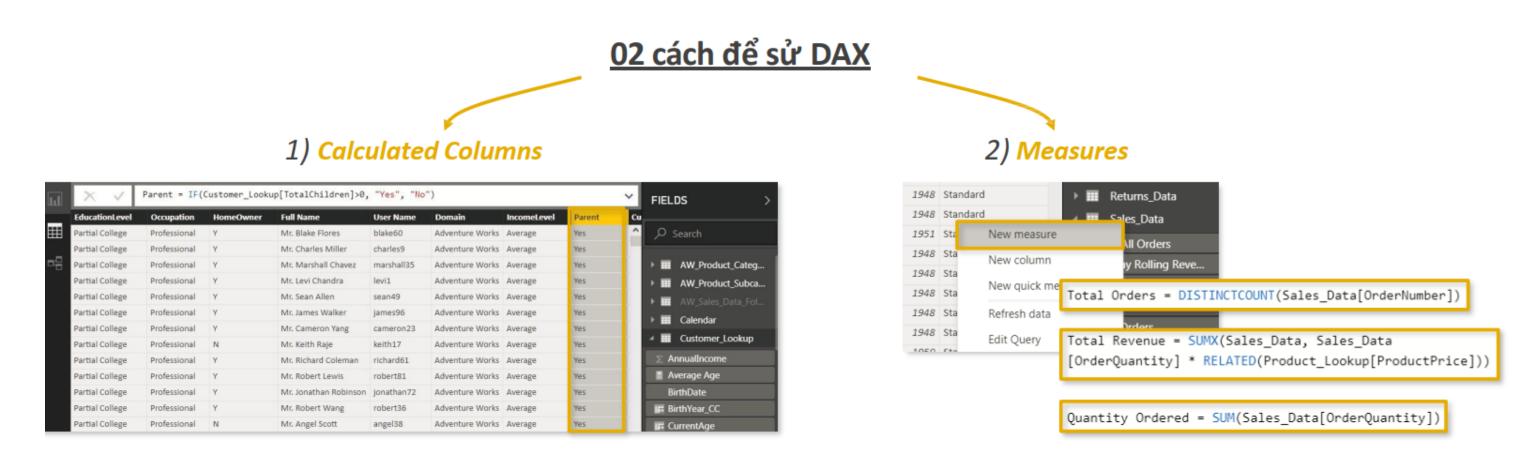
02. Các hàm DAX phổ biến



### Data Model là gì?

**DAX** là viết tắt của cụm từ Data Analysis Expressions là tập hợp các hàm, toán tử và hằng số, được sử dụng để giải quyết những bài toán từ cơ bản đến phức tạp. Từ đó, trả về một hoặc nhiều giá trị từ dữ liệu có sẵn.

- Một cách dễ hiểu, DAX được coi là hàm trong Power BI.
- Tuy nhiên, vượt xa khả năng của các hàm thuần túy, với các chức năng được xây dựng chuyên biệt để làm việc với các data model, DAX rất mạnh mẽ và linh hoạt.





# 3 yếu tố của DAX

Bạn cần biết rằng có rất nhiều yếu tố quan trọng khác nhưng đây là 3 vấn đề quan trọng mà nhất định phải nắm. Đó là Syntax (cú pháp), Function (chức năng) và Context (ngữ cảnh).

- Syntax có thể hiểu là cú pháp, trước khi tạo ra các công thức tính toán, bạn cần biết cú pháp của Power BI.
- Function được định nghĩa là các công thức được xác định trước, thực hiện những phép tính bằng các giá trị cụ thể được gọi là đối số, theo một thứ tự hoặc cấu trúc cụ thể.
- Context là ngữ cảnh sử dụng DAX. Có 2 loại Context trong DAX đó là Row Context và Filter Context.
  - Row Context áp dụng bất cứ khi nào một công thức có một hàm áp dụng bộ lọc để xác định một hàng trong bảng.
  - Filter Context giống như là một hoặc nhiều bộ lọc được áp dụng trong một phép tính mà được xác định kết quả hoặc giá trị.



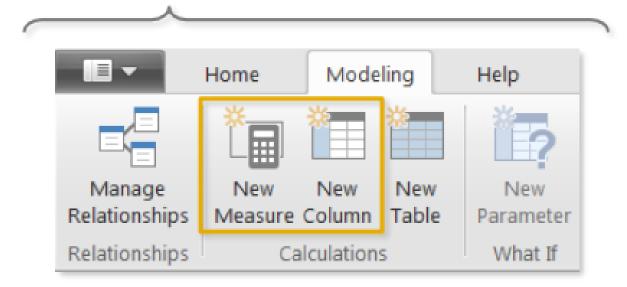
So sánh DAX (Power BI) và Hàm (Excel)

DAX	HÀM
<ul> <li>Thường quy định kiểu dữ liệu đầu vào</li> <li>Chỉ tham chiếu theo Cột hoặc Bảng</li> <li>Kết quả trả về có thể là một Giá trị, một Bảng</li> <li>DAX yêu cầu tạo ra các mối quan hệ giữa các bảng để sử dụng.</li> </ul>	<ul> <li>Thường không quy định, nếu sai thì báo lỗi</li> <li>Tham chiếu theo Ô - vùng và cả Cột - bảng</li> <li>Kết quả trả về là một Giá trị</li> <li>Có thể làm việc với các bảng độc lập</li> </ul>



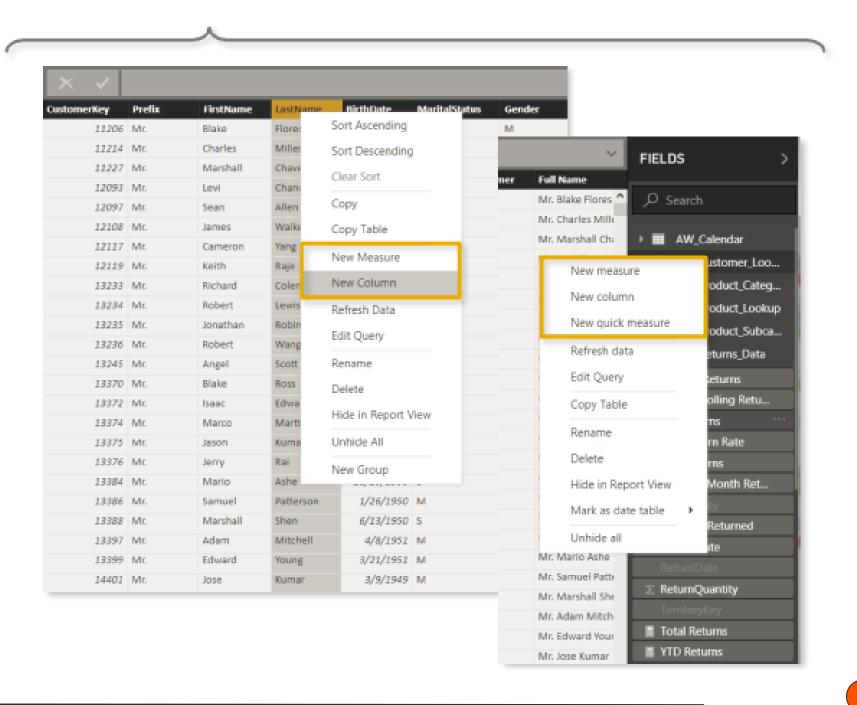
### **Thêm Columns & Measures**

Option 1: Chọn "New Measure" hoặc "New Column" trong tab Modeling

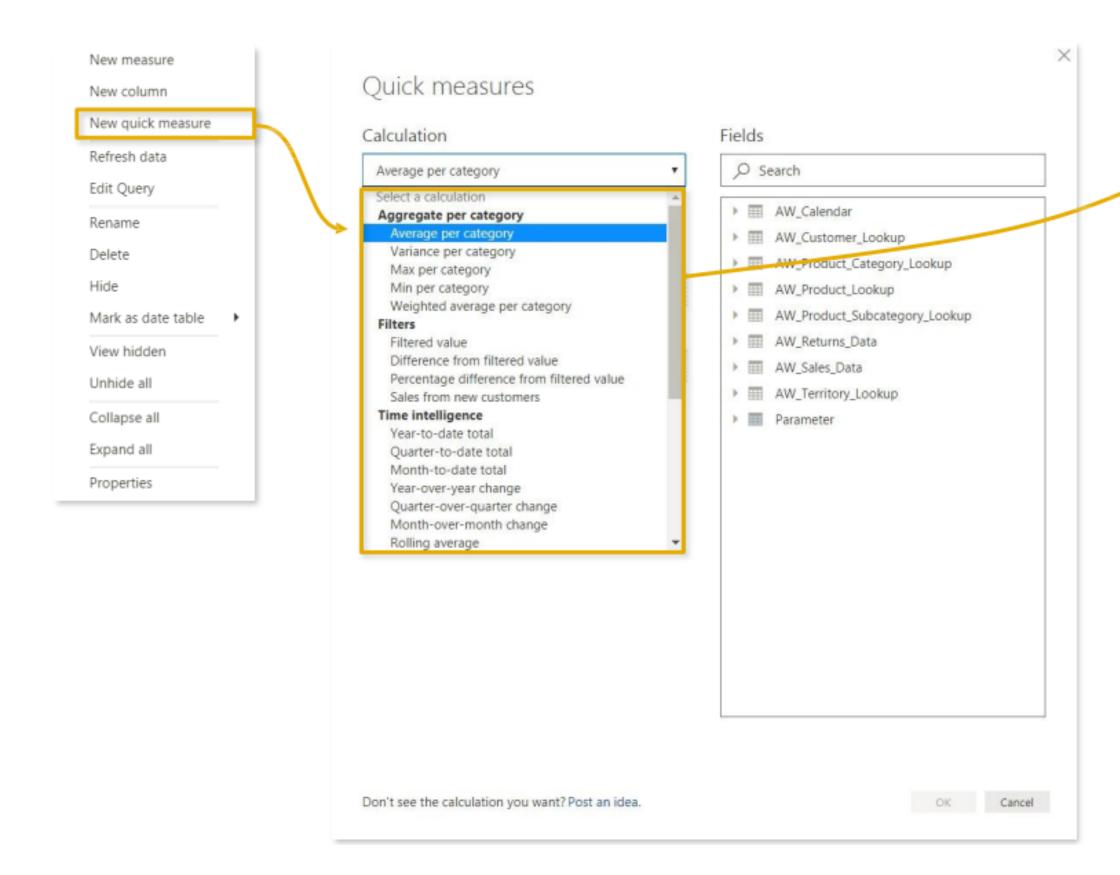


Khi bạn chèn một Columns hoặc Measures sử dụng tab **Modeling**, PowerBI sẽ mặc định bảng đang chọn làm dữ liệu tham chiếu, hoặc bảng đầu tiêu trong list thông thường

 Measures có thể chọn lại tables (trong phần "Properties" options trong tab Modeling), nhưng Option 2 sẽ cho phép bạn có sự chủ động hơn về vị trí và về bảng Option 2: Chuột phải trong table (trong Data view) hoặc trong Field List (cả Data và Report view)



# **Quick Measures**



Quick Measures là các công thức dựng trước dạng templates, cho phép bạn kéo và thả vào các trường, hơn là phải viết DAX từ đầu

Mặc dù công cụ này khá hữu ích khi sử dụng, đặc biệt là với các hàm có tính chất phức tạp (như weighted averages hay time intelligence...), nhưng nó sẽ khiến bạn lười hơn và đôi khi không hiểu rõ về DAX



### NOTE:

Just say **"NO"** với quick measures



### **Calculated Columns**

Calculated Columns cho phép bạn thêm mới cột vào data, dựa trên các biểu thức tính toán từ các côt đã có sẵn.

- DAX tham chiếu tới các trường theo cột và bảng, không tham chiếu theo ô hay vùng.
- Calculated Columns tạo ra các giá trị trong tables
- Calculated Columns rất tốt để tạo ra các cột giá trị mới dựa trên các dữ liệu tham chiếu từ hàng, nhưng lại không hữu dụng cho việc tính toán để lên trực tiếp báo cáo.

### NOTE:

• Thông thường, calculated columns sử dụng khi bạn muốn tạo cột trong bảng (một phần của Query Editor)

# **Measures**

Measures là công thức DAX sử dụng để tính toán giá trị.

- Giống như calculated columns, measures cũng tham chiếu tới tables và columns.
- Tuy nhiên, measure không hiện hữu giá trị trong bảng; nó chỉ có thể "nhìn thấy" khi visualize như khi sử dụng chart hay matrix (giống như một trường tính toán trong Excel pivot)
- Measures tính toán dựa trên các dữ liệu lọc từ bảng, nghĩa là nó sẽ cập nhật lại khi các trường hoặc dữ liệu lọc từ bảng thay đổi.

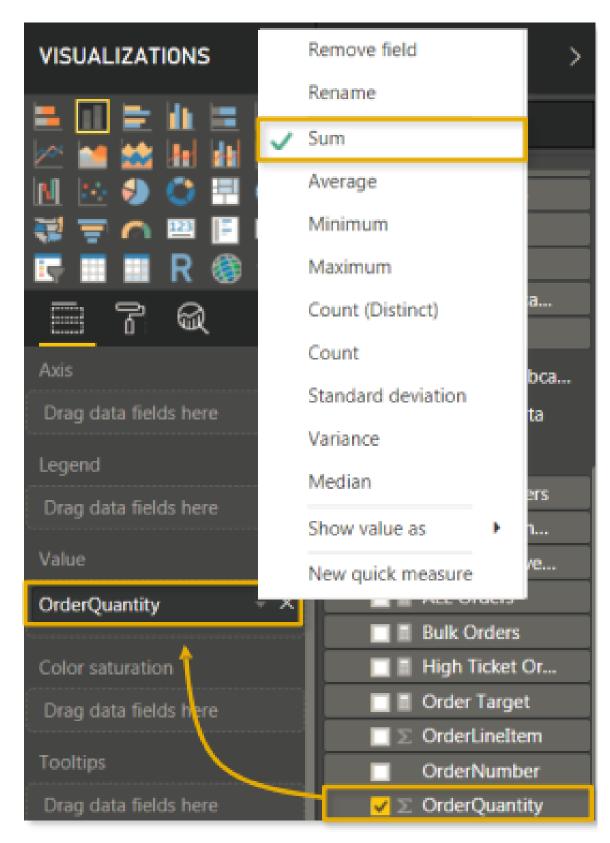
### NOTE:

• Thông thường, measures dùng khi một hàng đơn lẻ không đủ cho bạn một kết quả (nói cách khác, khi bạn cần tổng hơp dữ liệu)

 $(\gt$ 

# **Measures**

- Measures ẩn là các measures được tạo ra khi bạn kéo các trường dữ liệu số (như "OrderQuantity") trong trình tạo visual và chọn một dạng tính toán nào đó (Sum, Average, Min, Max, ...)
- Measures hiện là các measures được tạo ra bằng việc nhập các hàm DAX (hoặc thêm một "quick measures") để xác định giá trị.



Ví dụ một implicit measure





### Cấu trúc của DAX

### TÊN MEASURE

 Note: Measures luôn ở trong dấu ngoặc (vd: [Total Quantity]) khi tham chiếu trong công thức, do vậy, có thể sử dụng dấu cách (space) trong tên measures

Tên TABLE Tên COLUMN
tham chiếu
ây, có thể sử dụng
g tên measures

Tôtal Quantity: =SUM(Transactions[quantity])

### **HÀM (Function)**

- Không phải lúc nào calculated comlumn cũng sử dụng các hàm, nhưng measure thì luôn luôn:
  - Trong một Calculated Column, hàm
     =Transactions[quantity] sẽ trả về giá trị là số
     lượng giao dịch cho từng hàng (ô) trong cột
  - Trong một Measure, hàm
     =Transactions[quantity] sẽ trả về một giá trị error
     do Power BI không thể dịch hiểu như là một giá
     trị riêng lẻ



Loại số học	Ý nghĩa	Ví dụ
+	Cộng	2 + 7
-	Trừ	5-3
*	Nhân	2 * 6
/	Chia	4/2
^	Mũ	2 ^ 5

Loại so sánh	Ý nghĩa	Ví dụ
=	Bằng	[City]="Boston"
>	Lớn hơn	[Quantity]>10
<	Nhỏ hơn	[Quantity]<10
>=	Lớn hơn hoặc bằng	[Unit_Price]>=2.5
<=	Nhỏ hơn hoặc bằng	[Unit_Price]<=2.5
<b>&lt;&gt;</b>	Khác	[Country]<>"Mexico"

### Chú ý tới 02 biểu thức này

Loại Text/Logic		Ý nghĩa	Ví dụ
	&	Nối 02 giá trị dạng text thành 01 giá trị	[City] & " " & [State]
	&&	Tạo một điều kiện VÀ giữa 02 biểu thức logic	([State]="MA") && ([Quantity]>10)
	(double pipe)	Tạo một điều kiện HOẶC giữa 02 biểu thức logic	([State]="MA")    ([State]="CT")
	IN	Tạo một điều kiện HOẶC logic dựa trên danh sách đã cho (sử dụng dấu ngoặc)	'Store Lookup'[State] IN { "MA", "CT", "NY" }



### Các nhóm hàm phổ biến trong DAX

# MATH & STATS Functions

Hàm tính toán cơ bản cũng như tính toán trên vong lặp sử dụng ngữ cảnh hàng

#### Ví dụ:

- SUM
- AVERAGE
- MAX/MIN
- DIVIDE
- COUNT/COUNTA
- COUNTROWS
- DISTINCTCOUNT
- SUMX
- AVERAGEX
- MAXX/MINX
- RANKX
- COUNTX

# **LOGICAL** Functions

Hàm trả về thông tin về các giá trị trong biểu thức dựa trên điều kiện đã cho

#### Ví dụ:

- ) IF
- IFERROR
- AND
- OR
- NOT
- SWITCH
- TRUE
- FALSE

# **TEXT** Functions

Các hàm để thao tác các chuỗi văn bản hoặc các định dạng điều khiển cho ngày, giờ hoặc số

#### Ví dụ:

- CONCATENATE
- FORMAT
- LEFT/MID/RIGHT
- UPPER/LOWER
- PROPER
- LEN
- SEARCH/FIND
- REPLACE
- REPT
- SUBSTITUTE
- TRIM
- UNICHAR

# **FILTER** Functions

Các hàm tra cứu dựa trên các bảng có liên quan và các hàm lọc để tính toán

#### Ví dụ:

- CALCULATE
- FILTER
- ALL
- ALLEXCEPT
- RELATED
- RELATEDTABLE
- DISTINCT
- VALUES
- EARLIER/EARLIEST
- HASONEVALUE
- HASONEFILTER
- ISFILTERED
- USERELATIONSHIP

# DATE & TIME Functions

Các chức năng ngày và giờ cơ bản cũng như các hàm thời gian nâng cao

#### Ví dụ:

- DATEDIFF
- YEARFRAC
- YEAR/MONTH/DAY
- HOUR/MINUTE/SECOND
- TODAY/NOW
- WEEKDAY/WEEKNUM

#### Time Intelligence:

- DATESYTD
- DATESQTD
- DATESMTD
- DATEADD
- DATESINPERIOD



### Hàm DATE & TIME cơ bản

DAY/MONTH/ YEAR()

Trả về ngày trong tháng (1-31), tháng trong năm (1-12) hoặc năm của một ngày

=DAY/MONTH/YEAR(Date)

HOUR/MINUTE/ SECOND()

Trả về giờ (0-23), phút (0-59) hoặc giây (0-59) của một giá trị thời gian đã cho

=HOUR/MINUTE/SECOND(Datetime)

TODAY/NOW()

Trả về ngày hiện tại hoặc thời gian chính xác

=TODAY/NOW()

WEEKDAY/ WEEKNUM()

Trả về số ngày trong tuần từ 1 (Chủ nhật) đến 7 (Thứ bảy) hoặc tuần # của năm

=WEEKDAY/WEEKNUM(Date, [ReturnType])

**EOMONTH()** 

Trả về ngày của ngày cuối cùng của tháng, +/- một số tháng được chỉ định

=EOMONTH(StartDate, Months)

DATEDIFF()

Trả về chênh lệch giữa hai ngày, dựa trên khoảng thời gian đã chọn

=DATEDIFF(Date1, Date2, Interval)



# Hàm LOGICAL Cơ bản (IF, AND, OR)

IF()

Kiểm tra nếu một điều kiện nhất định được đáp ứng và trả về một giá trị nếu điều kiện là TRUE và một điều kiện khác nếu điều kiện là FALSE

=IF(LogicalTest, ResultIfTrue, [ResultIfFalse])

IFERROR()

Đánh giá một biểu thức và trả về một giá trị được chỉ định nếu biểu thức trả về lỗi, nếu không thì trả về chính biểu thức

=IFERROR(Value, ValueIfError)

AND()

Kiểm tra xem cả hai đối số có TRUE không và trả về TRUE nếu cả hai đối số đều TRUE, nếu không thì trả về FALSE

=AND(Logical1, Logical2)

Lưu ý: Sử dụng && và || toán tử nếu bạn muốn bao gồm nhiều hơn hai điều kiện!

OR()

Kiểm tra xem một trong các đối số có TRUE để trả về TRUE không và trả về FALSE nếu cả hai đối số là FALSE

=OR(Logical1, Logical2)



### Hàm TEXT cơ bản

LEN()

Trả về số lượng ký tự trong một chuỗi

=LEN(Text)

Lưu ý: Sử dụng toán tử & như một phím tắt hoặc để kết hợp nhiều hơn hai chuỗi!

**CONCATENATE()** 

Nối hai chuỗi văn bản thành một

=CONCATENATE(Text1, Text2)

LEFT/MID/ RIGHT()

Trả về một số ký tự từ đầu / giữa / cuối của chuỗi văn bản =LEFT/RIGHT(Text, [NumChars])

=MID(Text, StartPosition, NumChars)

UPPER/LOWER/ PROPER() Chuyển đổi các chữ cái trong một chuỗi thành trường hợp trên / dưới / thích hợp

=UPPER/LOWER/PROPER(Text)

**SUBSTITUTE()** 

Thay thế một thể hiện của văn bản hiện tại bằng văn bản mới trong một chuỗi =SUBSTITUTE(Text, OldText, NewText, [InstanceNumber])

SEARCH()

Trả về vị trí nơi một chuỗi được chỉ định hoặc ký tự được tìm thấy, từ trái sang phải =SEARCH(FindText, WithinText, [StartPosition], [NotFoundValue])



# RELATED()

Trả về các giá trị liên quan theo hàng của bảng dựa trên **relationships** với các bảng khác

# =RELATED(ColumnName)

Cột chứa các giá trị bạn muốn truy xuất Examples:

- Product\_Lookup[ProductName]
- Territory\_Lookup[Country]

### NOTE!



RELATED hoạt động gần như hàm VLOOKUP – nó sử dụng mối quan hệ giữa các bảng để đẩy dữ liệu từ một bảng sang một bảng khác như là một cột mới

Vì hàm này sử dụng **row context**, nên nó chỉ có thể sử dụng như một **calculated column** hoặc một phần của **iterator function** (FILTER, SUMX, MAXX, ...)



### TIP:

Tránh sử dụng RELATED để tạo ra một calculated columns trừ khi bạn cần thiết phải dùng, vì nó tăng đáng kể kích thước file; thay vào đó, nên dùng RELATED trong measure để FILTER hoặc SUMX



### Hàm MATH cơ bản

SUM()

Trả về kết quả là một tổng

=SUM(ColumnName)

AVERAGE()

Trả về trung bình (trung bình số học) của tất cả các số trong một cột

=AVERAGE(ColumnName)

MAX()

Trả về giá trị lớn nhất trong một cột hoặc giữa hai biểu thức vô hướng

=MAX(ColumnName) or =MAX(Scalar1, [Scalar2])

MIN()

Trả về giá trị nhỏ nhất trong một cột hoặc giữa hai biểu thức vô hướng

=MIN(ColumnName) or =MIN(Scalar1, [Scalar2])

DIVIDE()

Thực hiện phép chia và trả về kết quả thay thế (hoặc để trống) nếu div/0

=DIVIDE(Numerator, Denominator, [AlternateResult])



### **Hàm CALCULATE**

# CALCULATE()

Thực hiện một biểu thức hoặc công thức trong một điều kiện lọc

=CALCULATE(Expression, [Filter1], [Filter2],...)

Tên của một measure đã tồn tại, hoặc một công thức DAX
Ví dụ:

- [Total Orders]
- SUM(Returns\_Data[ReturnQuantity])

Các biểu thức lọc

(**Note:** yêu cầu đơn giản, giá trị cố định; không thể tạo filter trên một measures khác)

Ví dụ:

- Territory\_Lookup[Country] = "USA"
- Calendar[Year] > 1998



### TIP:

CALCULATE hoạt động tương tự **SUMIF** hoặc **COUNTIF** trong Excel, điểm khác biệt là nó có thể tính toán dựa trên mọi hàm đơn giản (không chỉ sum, count, ...); có thể gọi dễ hiểu hơn là **"CALCULATEIF"** 



ALL()

Trả về tất cả các hàng trong một bảng hoặc tất cả các giá trị trong một cột, bỏ qua mọi bộ lọc

# =ALL(Table or ColumnName, [ColumnName1], [ColumnName2],...)

Bảng hoặc cột muốn bỏ đi các điều kiện lọc

### **Examples:**

- Transactions
- Products[ProductCategory]

Danh sách cột chi tiết muốn bỏ đi điều kiện lọc

#### Notes:

- Nếu tham số đầu tiên của bạn là một bảng, bạn có thể chỉ định các cột bổ sung
- Tất cả các cột phải bao gồm tên bảng và đến từ cùng một bảng

### **Examples:**

- Customer\_Lookup[CustomerCity], Customer\_Lookup[CustomerCountry]
- Products[ProductName]



### TIP:

Thay cho việc thêm vào filter context, hàm ALL loại bỏ nó. Nó được sử dụng khi bạn muốn bỏ các filter context mà sẽ bị thay đổi các hàm liên quan



# FILTER()

Trả về một bảng biểu thị một tập hợp con của một bảng hoặc biểu thức khác

### =FILTER(Table, FilterExpression)

Tên Bảng để lọc

**Examples:** 

- Territory\_Lookup
- Customer\_Lookup

Biểu thức lọc, theo các hàng của bảng

### **Examples:**

- Territory\_Lookup[Country] = "USA"
- Calendar[Year] = 1998
- Products[Price] > [Overall Avg Price]

### NOTE!



FILTER được sử dụng để thêm một filter context mới, và có thể kiểm soát các điều kiện lọc phức tạp này tốt hơn cho CALCULATE

Vì FILTER trả về kết quả là một bảng, nó hay được sử dụng để tạo ra các *input* cho hàm



### TIP:

Vì FILTER đọc dữ liệu theo từng hàng trong bảng, nó có thể cho một tốc độ xử lý chậm hơn; không sử dụng FILTER nếu hàm CALCULATE làm được chức năng tương tự



Hàm (or "X") cho phép bạn lặp qua tính toán hoặc biểu thức tương tự trên mỗi hàng của một bảng và sau đó áp dụng một số công thức (SUM, MAX, v.v.)

=SUMX(Table, Expression)

Tên hàm

Bảng mà trong đó công thức

Sẽ chạy

Examples:

Examples:

- SUMX
- COUNTX
- AVERAGEX
- RANKX
- MAXX/MINX

- Sales
- FILTER(Sales, RELATED(Products[Category])="Clothing")

- [Total Orders]
- Sales[RetailPrice] \* Sales[Quantity]



### TIP:

Tưởng tượng rằng hàm này sẽ thêm một cột tạm thời trong bảng, tính toán các giá trị trong mỗi hàng (dựa trên biểu thức) và sau đó tổng hợp giá trị (giống như cách SUMPRODUCT hoạt động)



# Hàm TIME INTELLIGENCE

Các hàm Time Intelligence cho phép bạn dễ hàng tính toán với các so sánh thời gian phổ biến trong phân tích dữ liệu:

To-Date

Performance = CALCULATE(Measure, DATESYTD(Calendar[Date]))

Sử dụng DATESQTD cho Qúy hoặc DATESMTD cho Tháng

**Previous** Period

=CALCULATE(Measure, DATEADD(Calendar[Date], -1, MONTH))

Running Total

=CALCULATE(Measure,

Chọn một kiểu thời gian (DAY, MONTH, QUARTER, or YEAR) và

chọn mốc so sánh (ví dụ previous month, quay lại 10ngày ...)

DATESINPERIOD(Calendar[Date], MAX(Calendar[Date]), -10, DAY))