

MASTERING DAX IN POWER BI



Group 3:
SO KIMLANG
TEANG LAKSEY
SUN SOKSEKA
EAV SARIN
SONGEAM KANHA

Contents

1

- AGGREGATION

2

- CONTEXT IN DAX

3

- DAX AND POWER QUERY



AGGREGATION

SUM

Adds all the numbers in a column.

```
SUM ( table[column] )
```

SUMX

Returns the sum of an expression evaluated for each row in a table.

```
SUMX (  
    Sales,  
    Sales[Price] * Sales[Quantity]  
)
```

City	Channel	Color	Size	Quantity	Price
Paris	Store	Red	Large	1	15
Paris	Store	Red	Small	2	13
Torino	Store	Green	Large	4	11
New York	Store	Green	Small	8	9
	Internet	Red	Large	16	7
	Internet	Red	Small	32	5
	Internet	Green	Large	64	3
	Internet	Green	Small	128	1

$$1 \times 15 = 15$$

$$2 \times 13 = 26$$

$$4 \times 11 = 44$$

$$8 \times 9 = 72$$

$$16 \times 7 = 112$$

$$32 \times 5 = 160$$

$$64 \times 3 = 192$$

$$128 \times 1 = 128$$

Result = 749

AVERAGE

Returns the average (arithmetic mean) of all the numbers in a column.

AVERAGEA

Returns the average (arithmetic mean) of the values in a column.

AVERAGEX

Calculates the average (arithmetic mean) of a set of expressions evaluated over a table.

1 Average Sales Per Month = `AVERAGE(Sales_Table[Sales])`

1 Measure = `AVERAGEA('Table'[Amount])`

Measure = `AVERAGEX('Table', 'Table'[Amount])`

COUNT

Counts the number of rows in the specified column that contain non-blank values

COUNTA

the number of rows in the specified column that contain non-blank values.COUNTA supports Boolean data type.

COUNTROWS

function counts the number of rows in the specified table, or in a table defined by an expression.

Number of sales id = **COUNT(Data1[Sales Id])**

COUNTA(<column>)

1 Number of sales id = **COUNTA(Data1[Sales Id])**

COUNTROWS([<table>])

Measure = **COUNTROWS(Data)**

MAX

Returns the largest value in a column, or between two scalar expressions.

Measure = MAX(<column>)

MAXA

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

MaxA = MAXA(Sheet[Order date])

MIN

Returns the smallest value in a column, or between two scalar expressions.

MINA

Logical values, such as TRUE and FALSE. Rows MAXA: Rows in the column that evaluates to logical values, such as TRUE and FALSE are treated as 1 if TRUE and 0 (zero) if FALSE.evaluate to TRUE count as 1; rows that evaluate to FALSE count as 0 (zero).

Measure =MIN(<column>)

MinA of date = MINA(Sheet[Order date])



CONTEXT IN DAX

CONTEXT

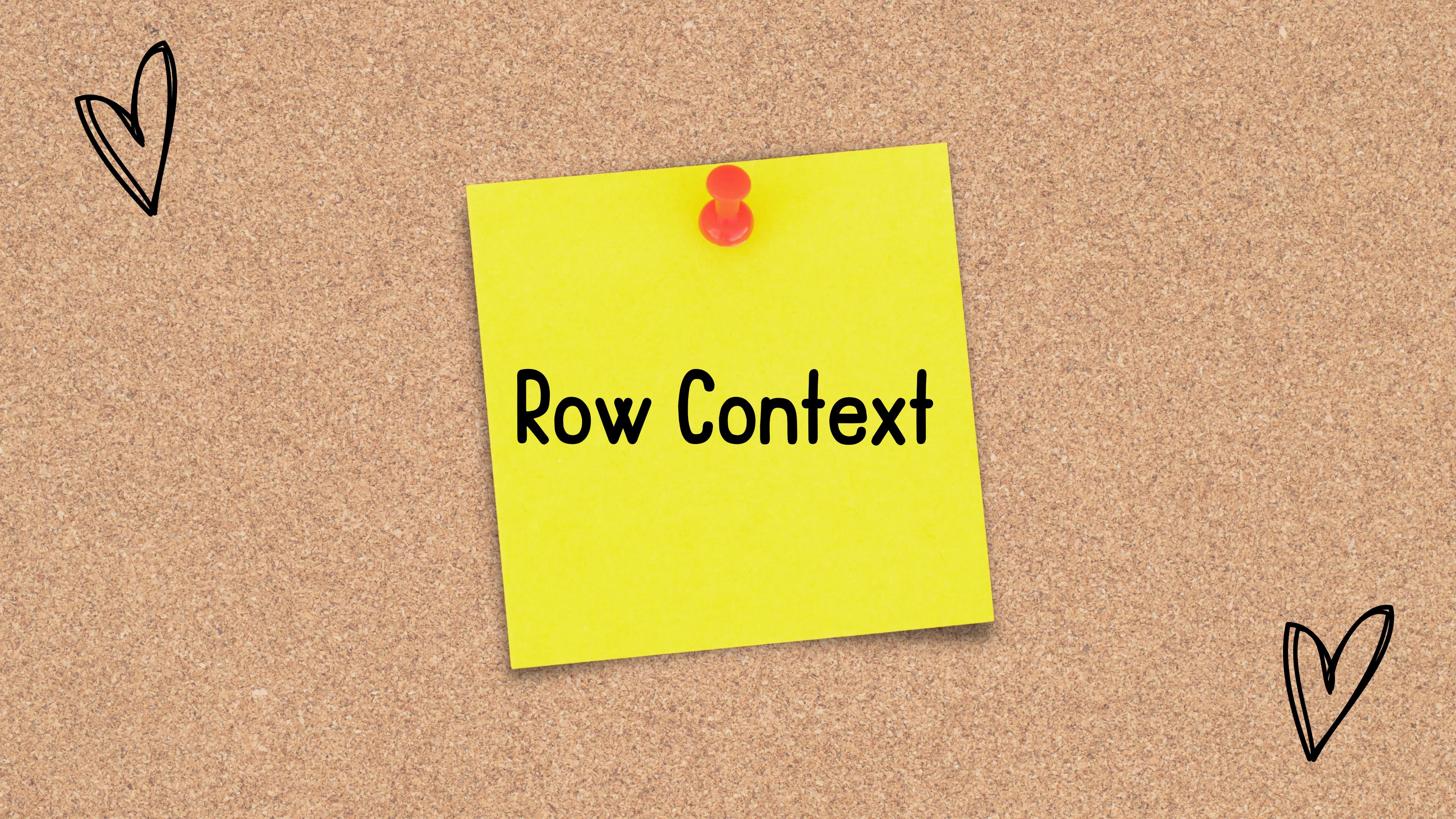
Context is what makes it possible to perform dynamic analysis. Understanding context is important for building and for troubleshooting formulas.

There are main types of context:

row context

filter context.





Row Context

Row Context in DAX

Row context can be thought of as "the current row." The Row Context allows DAX to consider each row individually when performing calculations. It evaluated one row at a time

Note

It is to be noted that here that SUMX is used instead to SUM

```
1 Cost Price = SUMX(Sales_Data, [Units_Sold] * [Unit_MakingCost])
```



Filter Context



Filter Context in DAX

Filter context defines the set of filters or criteria that are applied to a DAX calculation. These filters can come from various sources, such as slicers, filters, or calculated columns in your data model.

if you apply a filter for a specific date range, the SUM function would sum only the data within that date range.

Total Sales for Selected Year =
CALCULATE(
 SUM(Sales[SalesAmount]),
 FILTER(
 ALL('Date'),
 YEAR('Date'[Date]) = YEAR(SELECTEDVALUE('Date'[Date]))
)
)

Functions in Context in DAX

- RELATED

enables you to access data from related tables based on a specified relationship and filter context. This function is commonly used in scenarios involving one-to-many or many-to-one relationships.

`SUMX(Sales, RELATED(Products[Category]))`

- FILTER

The FILTER function is used to create a new table that filters another table or an expression in a specific context.

`SUMX(FILTER(Sales, YEAR(Sales[Date]) = 2023), Sales[SalesAmount])`

- ALL

function removes filters from a table or columns, restoring them to their original state before any filters were applied.

`CALCULATE(SUM(Sales[SalesAmount]), ALL(Sales))`

- ALLEXCEPT

function preserves some filters on a table or columns while removing all other filters.

`CALCULATE(SUM(Sales[SalesAmount]), ALLEXCEPT(Sales, Sales[ProductID]))`

Some Contexts of Filter Context in DAX

Row Context

When a formula is evaluated for a specific row, the filter context includes only the data related to that row

Filter Context

The filter context is created by applied filters on columns or measures.

Implicit Filter Context

Implicit filter context is automatically applied based on relationships between tables.

- SELECTED ROWS
- CALCULATE

- FILTER

- RELATED

Explicit Filter Context

It allows you to define specific conditions or criteria to filter data and modify the filter context for calculations.

Context Transition

It happens when you refer to a column or measure that is not part of the current row context.

- FILTER
- All
- CALCULATE



DAX AND POWER QUERY

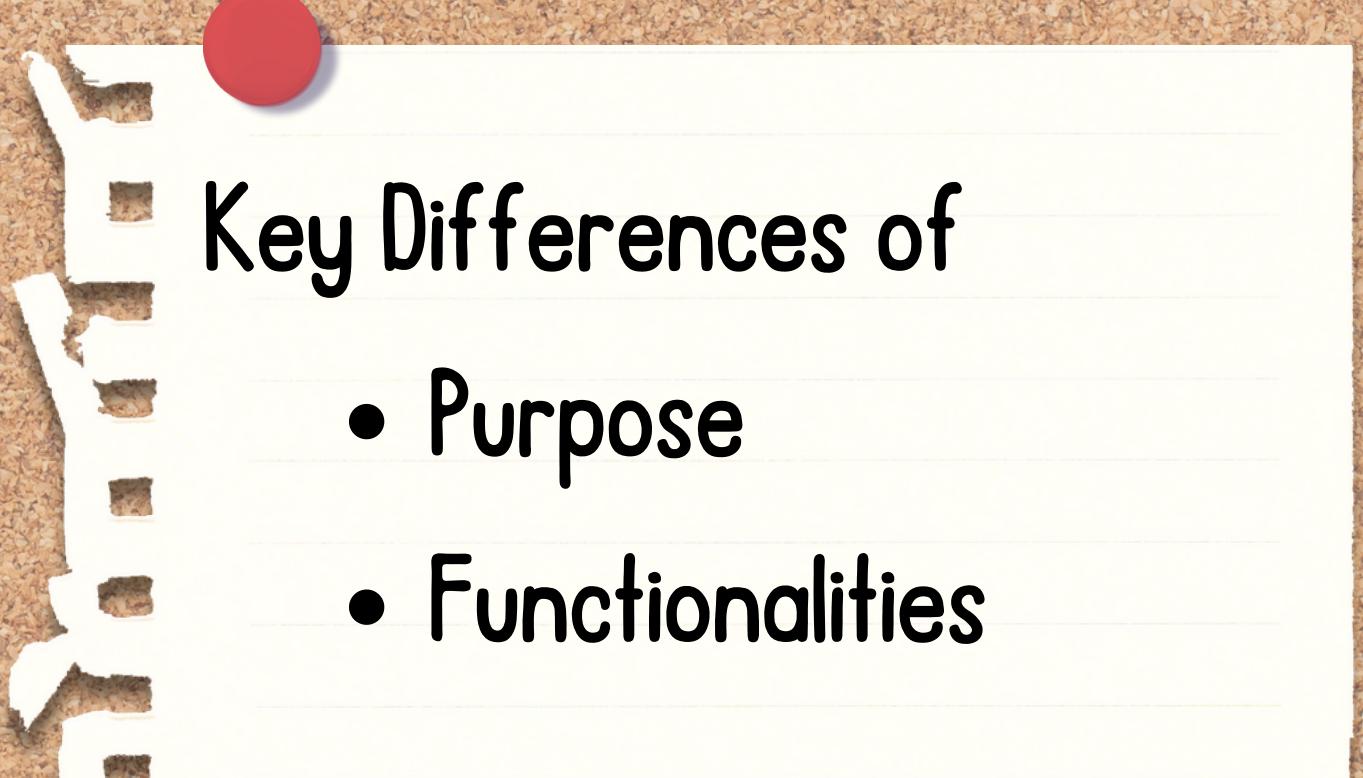
DAX AND POWER QUERY

The Differences in Data
Modeling

The Complementary in
Data Modeling

The Differences DAX Vs Power Query

- **DAX:** Data Analysis Expressions is a formula language that allows users to create calculated columns and measures in data model after loading data.
- **Power Query:** The M language is a data transformation language that allows users to clean and transform data before it is loaded into the data model.



- Purpose

DAX

- Calculation language
- Formula-cased
- Context-awareness
- Calculation engine
- Advance analytics

Power Query

- Data transformation
- Data sources connectivity
- Query editor
- Data cleansing
- Data loading

• Functionalities

DAX

- Calculated measures
- Aggregations and statistical functions
- Time Intelligence
- Filtering and slicers
- Calculated columns and tables
- Set up data modeling relationship

Power Query

- Data Import and connection
- Data transformation and cleansing
- Combining multiple files
- Data shaping and merging
- Building final queries before loading into memory



The Complementary in Data Modeling

- Power Query for Data Transformation
- DAX for calculations and Analysis
- Iterative Data Modeling process
- Performance Optimization

Thank you