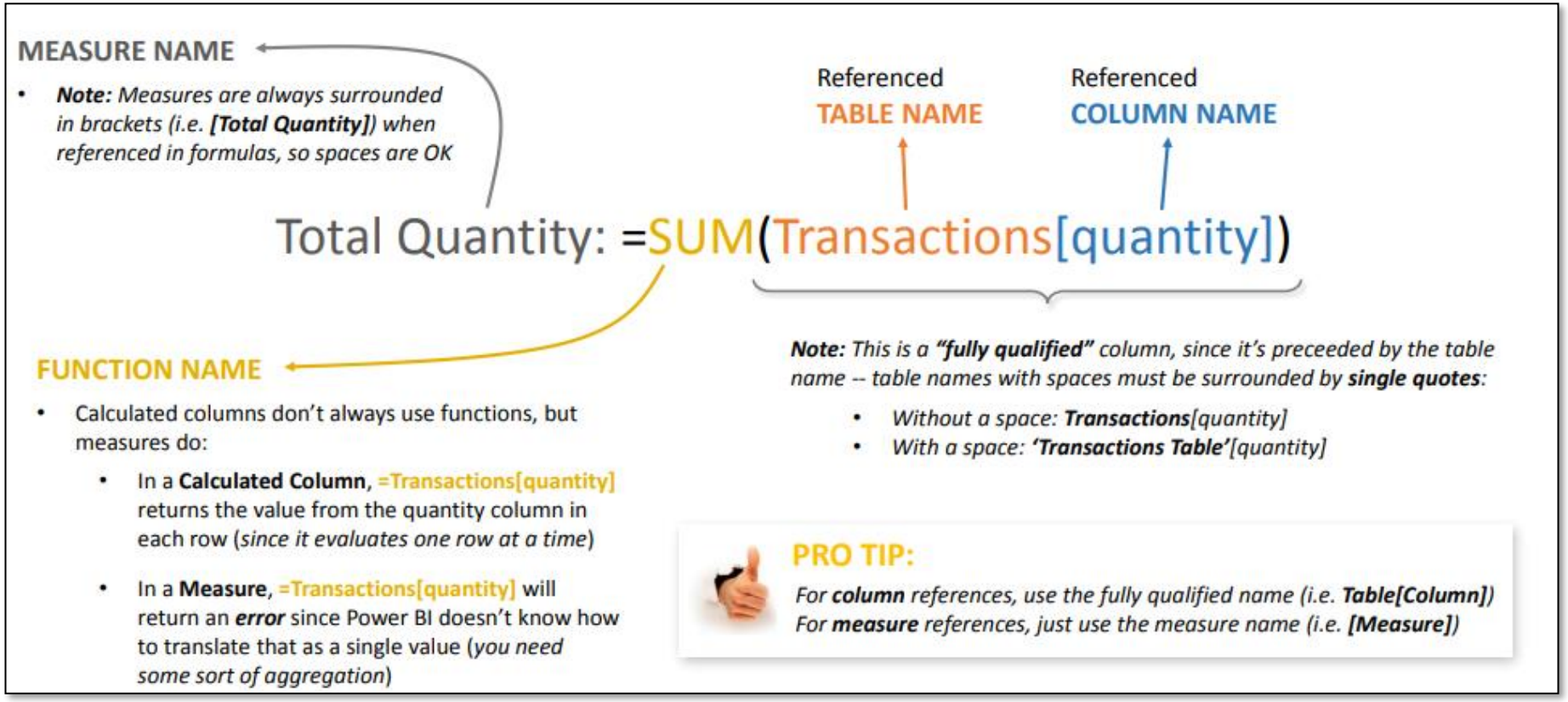
COMMON DAX FUNCTION

DAX Meet


MEASURE NAME
• Note: Measures are always surrounded in brackets (i.e. [Total Quantity]) when referenced in formulas, so spaces are OK
Referenced TABLE NAME
Referenced COLUMN NAME
Total Quantity: =SUM(Transactions[quantity])
FUNCTION NAME
• Calculated columns don't always use functions, but measures do:
• In a Calculated Column, =Transactions[quantity] returns the value from the quantity column in each row (since it evaluates one row at a time)
• In a Measure, =Transactions[quantity] will return an error since Power BI doesn't know how to translate that as a single value (you need some sort of aggregation)
Note: This is a "fully qualified" column, since it's preceeded by the table name -- table names with spaces must be surrounded by single quotes:
• Without a space: Transactions[quantity]
• With a space: 'Transactions Table'[quantity]
PRO TIP:
For column references, use the fully qualified name (i.e. Table[Column])
For measure references, just use the measure name (i.e. [Measure])

DAX Operators

Arithmetic operators

To perform basic mathematical operations such as addition, subtraction, or multiplication; combine numbers; and produce numeric results, use the following arithmetic operators.

Arithmetic operator	Meaning	Example
+ (plus sign)	Addition	3+3
– (minus sign)	Subtraction or sign	3–1–1
* (asterisk)	Multiplication	3*3
/ (forward slash)	Division	3/3
^ (caret)	Exponentiation	16^4

Comparison operators

You can compare two values with the following operators. When two values are compared by using these operators, the result is a logical value, either TRUE or FALSE.

Comparison operator	Meaning	Example
=	Equal to	[Region] = "USA"
==	Strict equal to	[Region] == "USA"
>	Greater than	[Sales Date] > "Jan 2009"
<	Less than	[Sales Date] < "Jan 1 2009"
>=	Greater than or equal to	[Amount] >= 20000
<=	Less than or equal to	[Amount] <= 100
<>	Not equal to	[Region] <> "USA"

Text concatenation operator

Use the ampersand (&) to join, or concatenate, two or more text strings to produce a single piece of text.

Text operator	Meaning	Example
& (ampersand)	Connects, or concatenates, two values to produce one continuous text value	[Region] & ", " & [City]

Logical operators

Use logical operators (&&) and (||) to combine expressions to produce a single result.

Text operator	Meaning	Examples
&& (double ampersand)	Creates an AND condition between two expressions that each have a Boolean result. If both expressions return TRUE, the combination of the expressions also returns TRUE; otherwise the combination returns FALSE.	([Region] = "France") && ([BikeBuyer] = "yes"))
|| (double pipe symbol)	Creates an OR condition between two logical expressions. If either expression returns TRUE, the result is TRUE; only when both expressions are FALSE is the result FALSE.	(([Region] = "France") || ([BikeBuyer] = "yes"))
IN	Creates a logical OR condition between each row being compared to a table. Note: the table constructor syntax uses curly braces.	'Product'[Color] IN { "Red", "Blue", "Black" }

*Head to DAX operators - DAX | Microsoft Learn for information about DAX syntax, operators, troubleshooting, etc.

Common Function Categories

MATH & STATS Functions

*Basic **aggregation** functions as well as "**iterators**" evaluated at the row-level*

Common Examples:

- SUM
- AVERAGE
- MAX/MIN
- DIVIDE
- COUNT/COUNTA
- COUNTROWS
- DISTINCTCOUNT

Iterator Functions:

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

LOGICAL Functions

*Functions for returning information about values in a given **conditional expression***

Common Examples:

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

TEXT Functions

*Functions to manipulate **text strings** or **control formats** for dates, times or numbers*

Common Examples:

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

FILTER Functions

***Lookup** functions based on related tables and **filtering** functions for dynamic calculations*

Common Examples:

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

DATE & TIME Functions

*Basic **date and time** functions as well as advanced **time intelligence** operations*

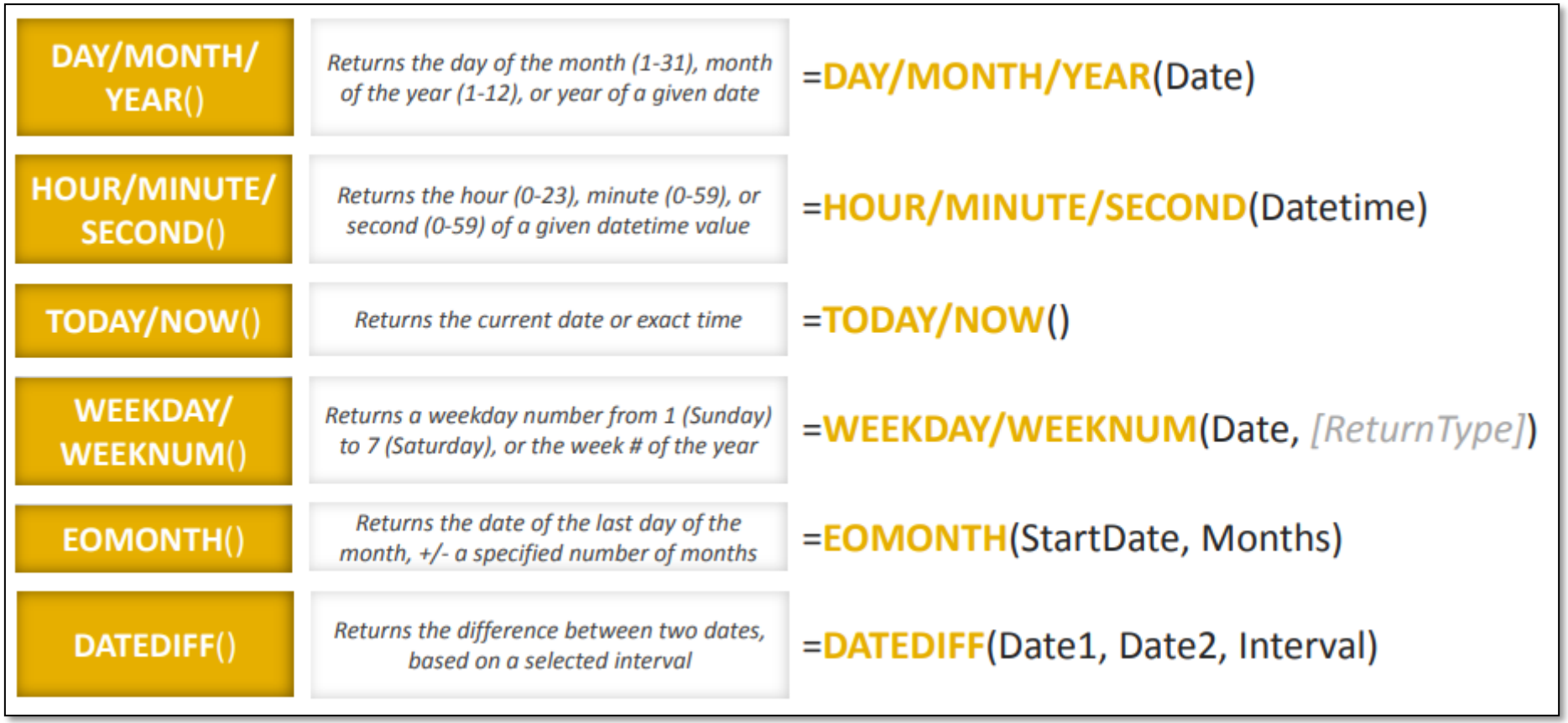
Common Examples:

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

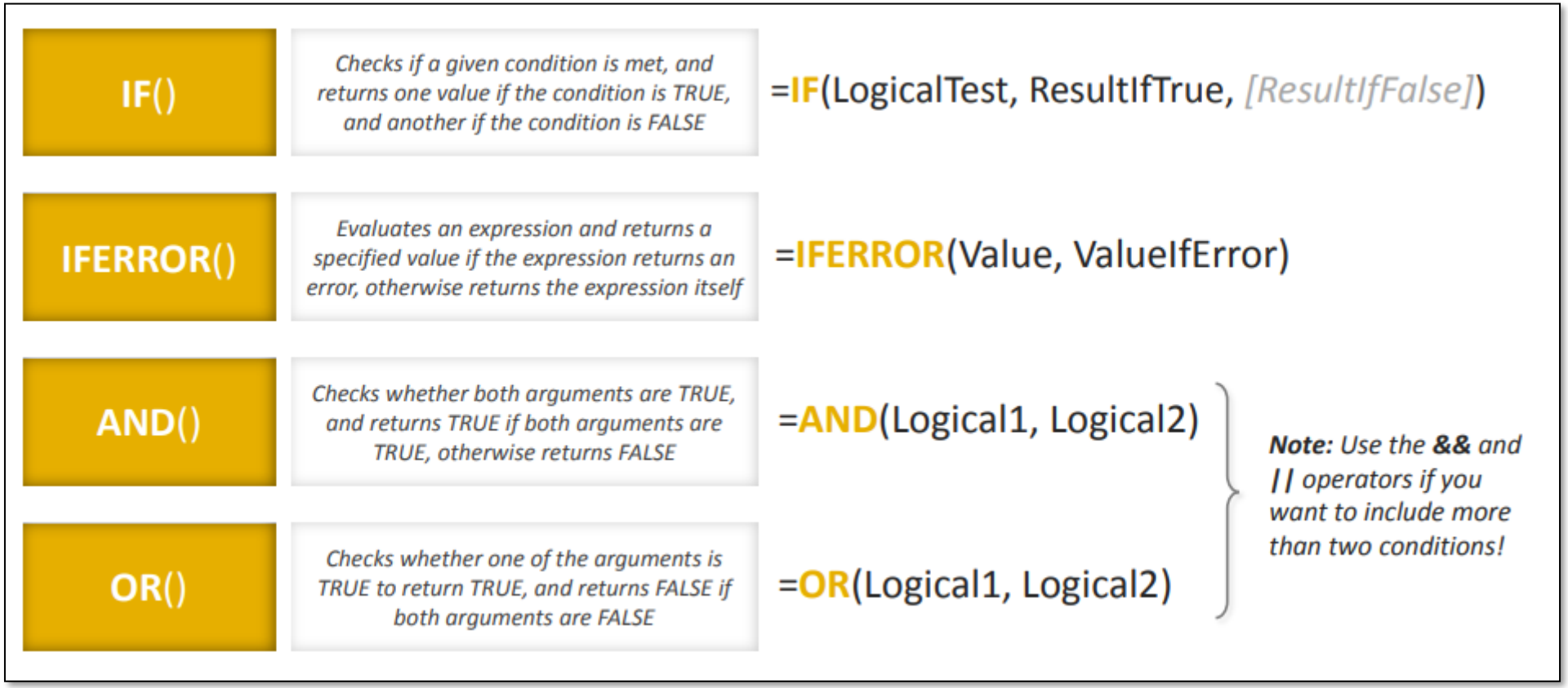
Time Intelligence Functions:

- DATESYTD
- DATESQTD
- DATESMTD
- DATEADD
- DATESINPERIOD

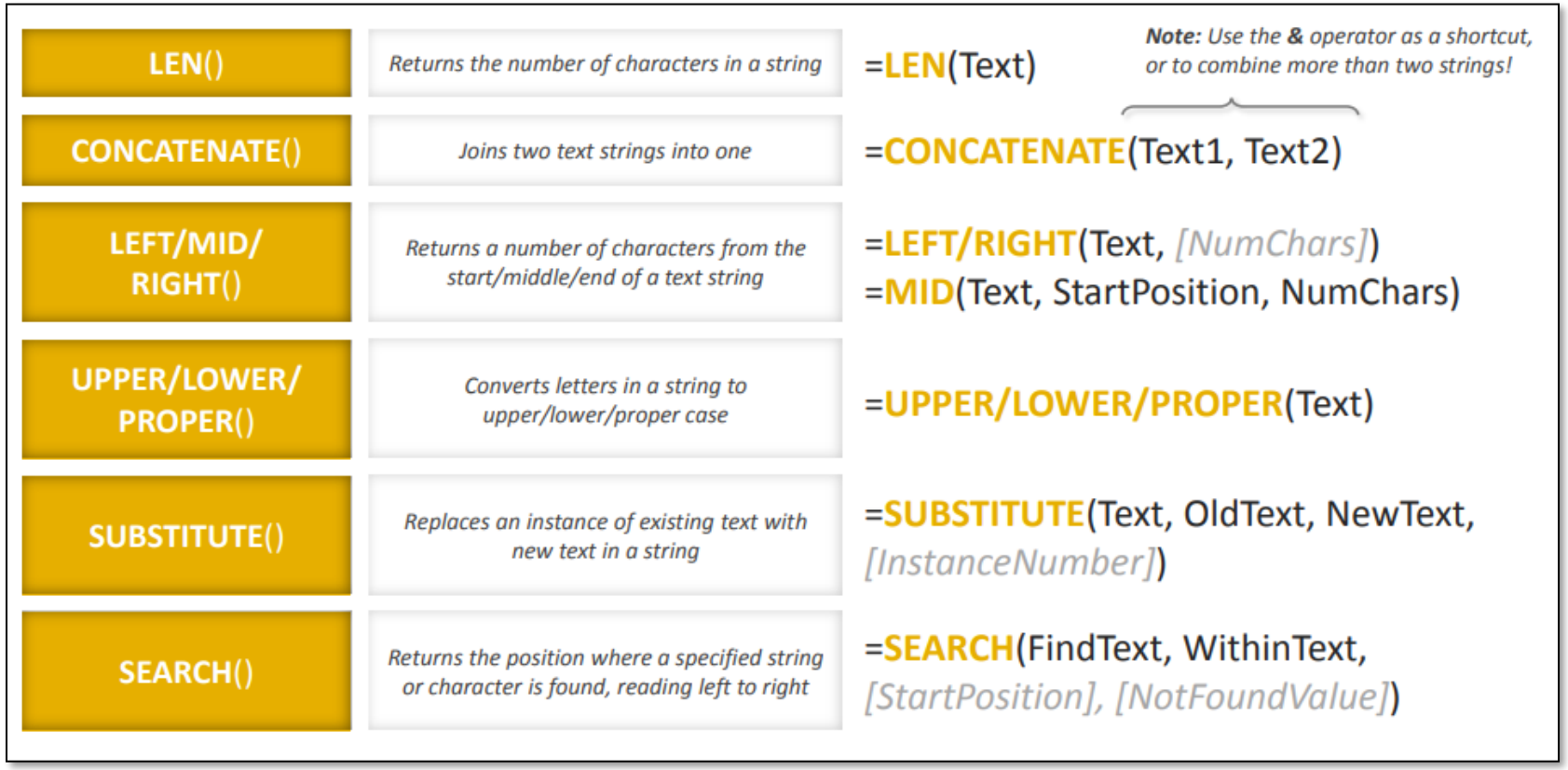
Basic Date & Time Function



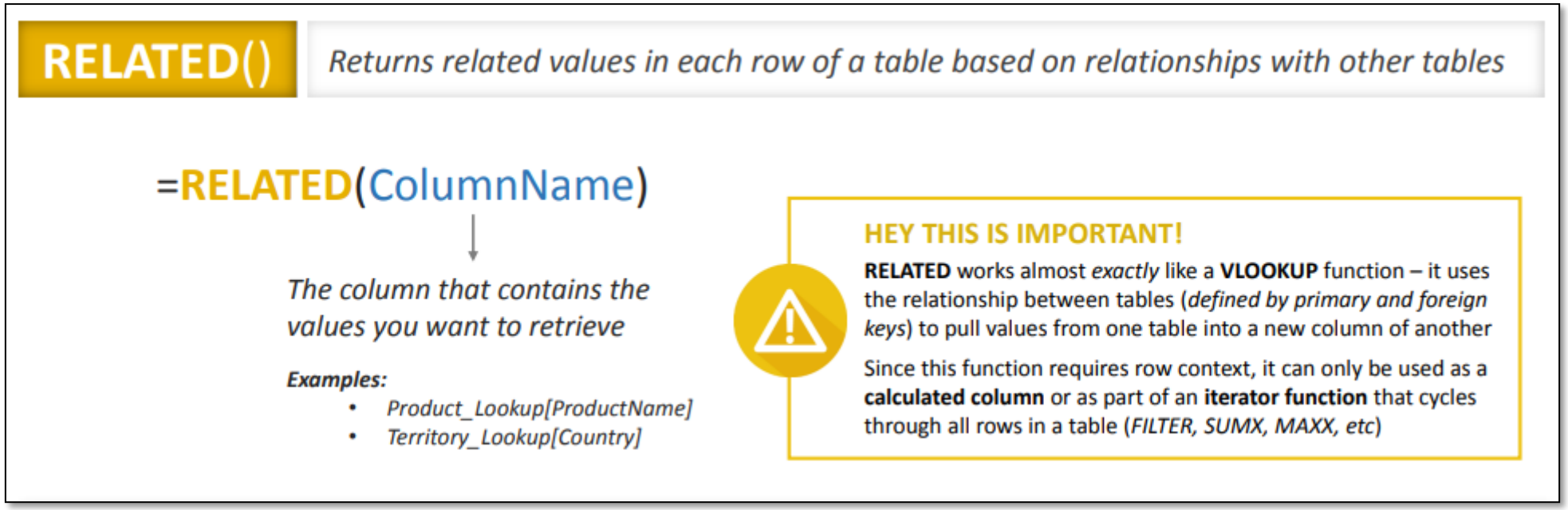
Basic Logical Functions (IF/AND/OR)



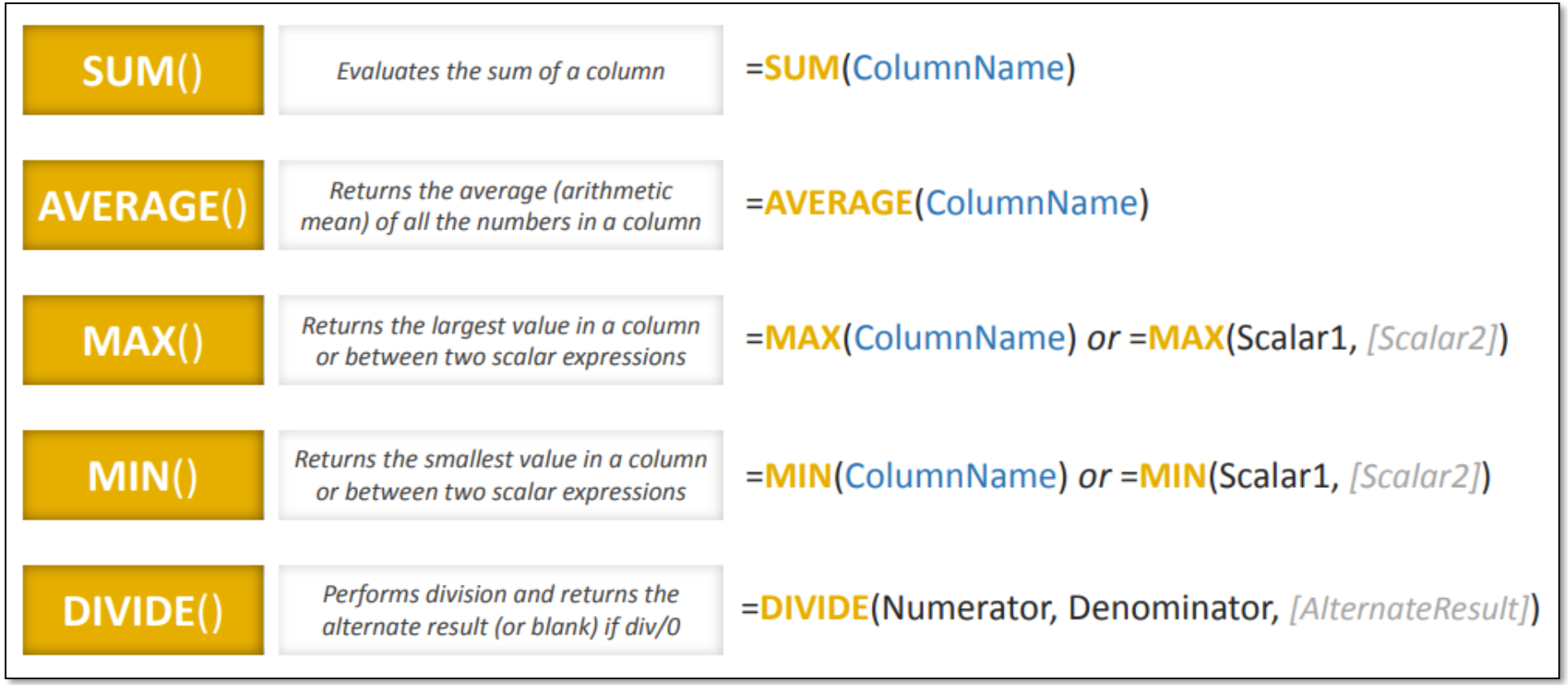
Text Function



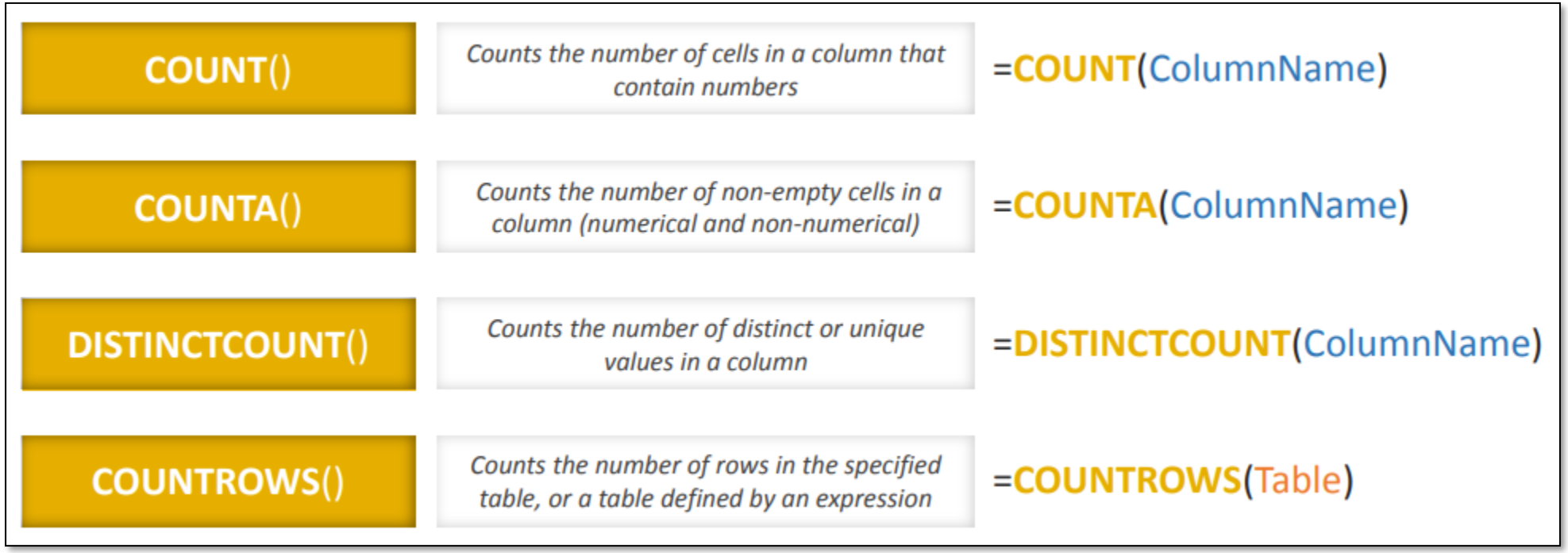
Related


RELATED()
Returns related values in each row of a table based on relationships with other tables
=RELATED(ColumnName)
The column that contains the values you want to retrieve
Examples:
Product_Lookup[ProductName]
Territory_Lookup[Country]
HEY THIS IS IMPORTANT!
RELATED works almost exactly like a VLOOKUP function – it uses the relationship between tables (defined by primary and foreign keys) to pull values from one table into a new column of another
Since this function requires row context, it can only be used as a calculated column or as part of an iterator function that cycles through all rows in a table (FILTER, SUMX, MAXX, etc)

Basic Math & Stats Functions



Count, Counta, Distinctcount & Countrows



Calculate

CALCULATE() — *Evaluates a given expression or formula under a set of defined filters*

=CALCULATE(Expression, *[Filter1], [Filter2],…*)

Name of an existing measure, or a DAX formula for a valid measure

Examples:

- *[Total Orders]*
- *SUM(Returns_Data[ReturnQuantity])*

*List of simple Boolean (True/False) filter expressions (**note:** these require simple, fixed values; you cannot create filters based on measures)*

Examples:

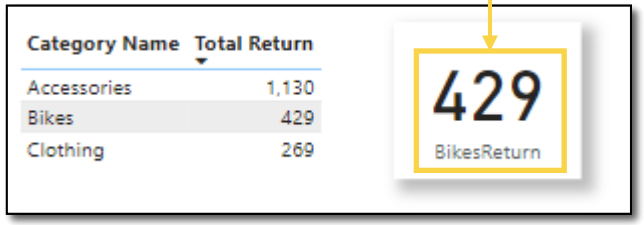
- *Territory_Lookup[Country] = "USA"*
- *Calendar[Year] > 1998*

PRO TIP:

*CALCULATE works just like **SUMIF** or **COUNTIF** in Excel, except it can evaluate measures based on ANY sort of calculation (not just a sum, count, etc); it may help to think of it like **"CALCULATEIF"***

Calculate (Example)

```
1  BikesReturn = CALCULATE([TotalReturn],AW_Product_Category_Lookup[CategoryName]="Bikes")
```



Here we've defined a new measure named "BikeReturns", which evaluates the "Total Returns" measure when the *CategoryName* in the Products table equals "Bikes"

Calculate Change the filter context

CALCULATE

Filters modified by CALCULATE

[CategoryName] = "Bikes"

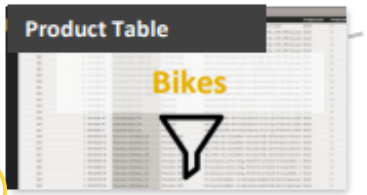
If the measure being evaluated contains a **CALCULATE** function, filter context is *overwritten* between **Step 1** & **Step 2**

STEP 1

Filter context is detected & applied

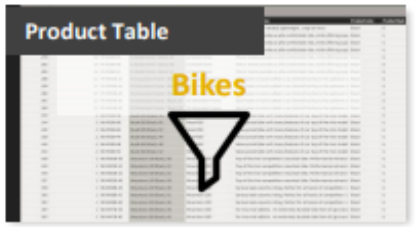
Category Name	Total Return	BikesReturn
Accessories	1,130	429
Bikes	429	429
Clothing	269	429
Components		429

Products[CategoryName] = "Accessories"



STEP 2

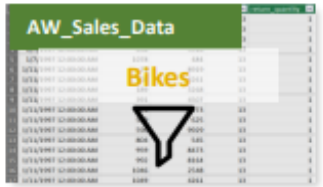
Filters flow "downstream" to all related tables

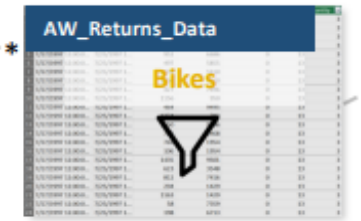


1 1

*

*





STEP 3

Measure formula evaluates against the filtered table

```
1 BikesReturn = CALCULATE([TotalReturn],AW_Product_Category_Lookup[CategoryName]="Bikes")
```

*Count of rows in the **AW_Returns_Data** table, filtered down to only rows where the product category is "Bikes"* = 429

All

ALL() *Returns all rows in a table, or all values in a column, ignoring any filters that have been applied*

=ALL(Table *or* ColumnName, *[ColumnName1], [ColumnName2],…*)

The table or column that you want to clear filters on

Examples:

- *Transactions*
- *Products[ProductCategory]*

List of columns that you want to clear filters on (optional)

Notes:

- *If your first parameter is a table, you can't specify additional columns*
- *All columns must include the table name, and come from the same table*

Examples:

- *Customer_Lookup[CustomerCity], Customer_Lookup[CustomerCountry]*
- *Products[ProductName]*

Filter

FILTER() *Returns a table that represents a subset of another table or expression*

=FILTER(Table, FilterExpression)

Table to be filtered

Examples:

- *Territory_Lookup*
- *Customer_Lookup*

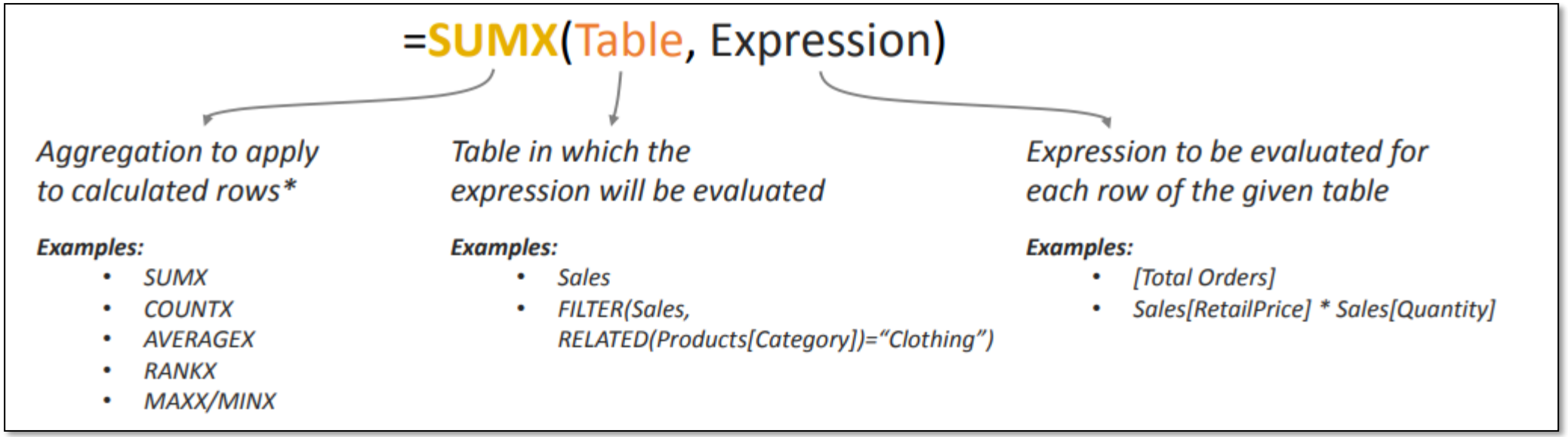
A Boolean (True/False) filter expression to be evaluated for each row of the table

Examples:

- *Territory_Lookup[Country] = "USA"*
- *Calendar[Year] = 1998*
- *Products[Price] > [Overall Avg Price]*

Iterator ("X") Functions

Iterator (or "**X**") **functions** allow you to loop through the same calculation or expression on each row of a table, and then apply some sort of aggregation to the results (SUM, MAX, etc)



Time Intelligence Formulas

Time Intelligence functions allow you to easily calculate common time comparisons: