Power BI - DAX - p1

DAX [Data Analysis Expression]

Calculated Columns

Row Context

Measures.

Filter Context

- Europe
- North America
- Pacific

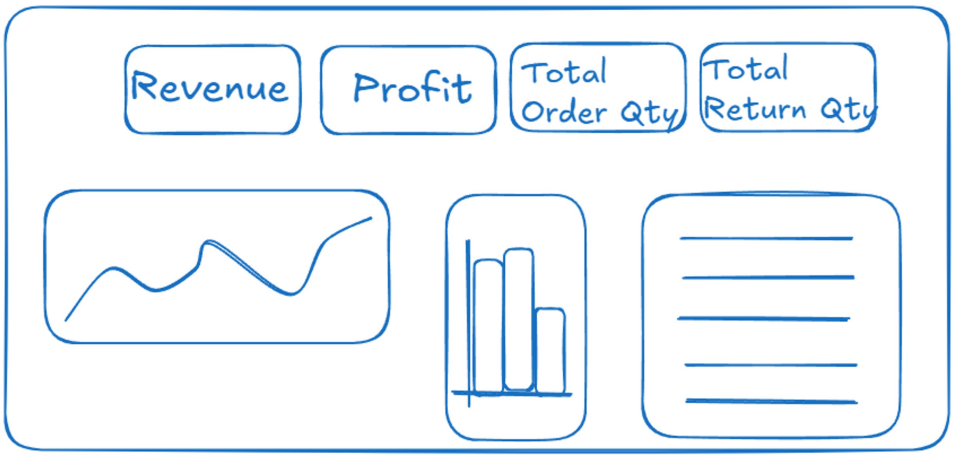
→ Revenue

Shark Tank

KPI , Matrices , Figures.

- Revenue Price * Total Order Qty.
- Profit

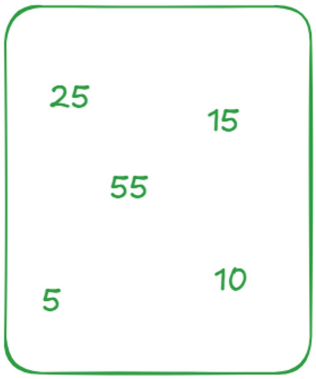
Variable rev_europe = Value;



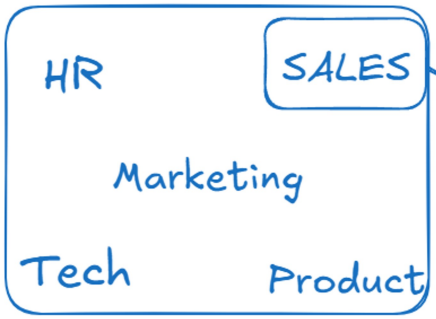
Total No. of Children	Is Married/ Not	Is Parent?
2	True	True
1	True	True
0	False	False
1	True	
2	True	
3	True	
0	False	

Row Context

- IFS()
- SUMIFS()
- COUNTIF()
- AVERAGEIF()
- MAXIF()



110 - SUM
110/5 = 22 - Average
55 - MAX
5 - MIN



Aggregation : Can't be perform without Grouping the data.

TableName[ColumnName]

```
Total Orders = DISTINCTCOUNT(Sales_Data[OrderNumber])
```

urns Data

```
Total Revenue = SUMX(Sales_Data, Sales_Data
[OrderQuantity] * RELATED(Product_Lookup[ProductPrice]))
```

TableName[ColumnName]

```
Quantity Ordered = SUM(Sales_Data[OrderQuantity])
```

Clothing

```
Bike Returns =
CALCULATE(
    [Total Returns],
    'Product Categories Lookup'[Category Name] = "Bikes"
)
```

Measure → Calling the existing measure

Filter

Category Name	Total Returns	Bike Returns
Accessories	1,115	
Bikes	427	427
Clothing	267	
Total	**1,809**	**427**

Clothing

Clothing Returns

267

CALCULATED FIELDS WITH DAX

In this section we'll use Data Analysis Expressions (DAX) to add calculated columns & measures to our model, and introduce topics like row & filter context, iterators and more

TOPICS WE'LL COVER:

1. DAX
2. Row & Filter Context
3. Common Functions
4. Iterators
5. Columns & Measures
6. DAX Syntax
7. Calculate
8. Time Intelligence

GOALS FOR THIS SECTION:

- Introduce DAX fundamentals and learn when to use calculated columns and measures.

- Understand the difference between row context and filter context, and how they impact DAX calculations.

- Learn DAX formula syntax, basic operators and common function categories (math, logical, text, date/time, filter, etc.).

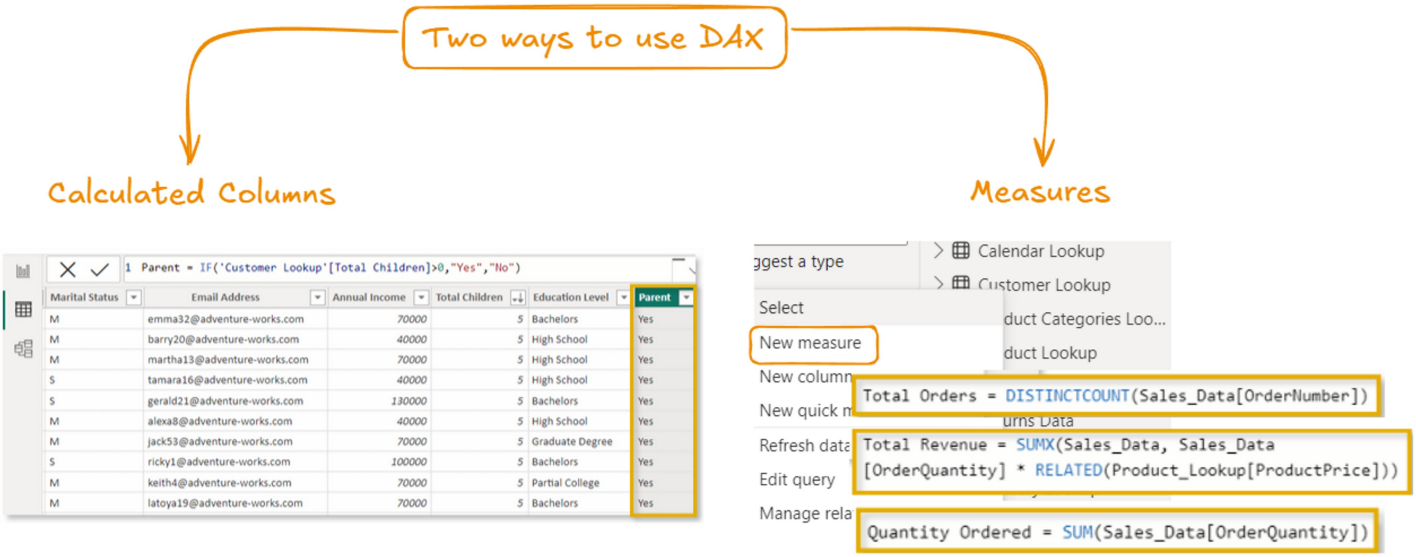
- Explore nested functions, and more complex topics like iterators and time intelligence patterns.

MEET DAX

Data Analysis Expressions (commonly known as DAX) is the formula language that drives the Power BI front-end. With DAX, you can:

- Go beyond the capabilities of traditional spreadsheet formulas, with powerful and flexible functions built specifically to work with relational data models.

- Add calculated columns (for filtering) and measures (for aggregation) to enhance data models.



M VS. DAX

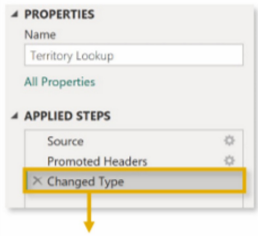
M and DAX are two distinct functional languages used within Power BI Desktop:

- M is used in the Power Query editor, and is designed specifically for extracting, transforming and loading data.

- DAX is used in the Power BI front-end, and is designed specifically for analyzing relational data models

M CODE

Query Editor:




```
#"Changed Type" = Table.TransformColumnTypes(    // Adding a new step
  #"Promoted Headers",                           // after we promoted headers
  {
    {"SalesTerritoryKey", Int64.Type},           // that changes column datatypes
    {"Region", type text},
    {"Country", type text},
    {"Continent", type text}
  }
)
```

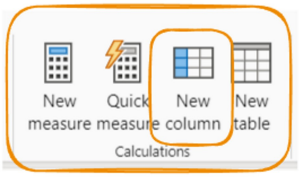
DAX

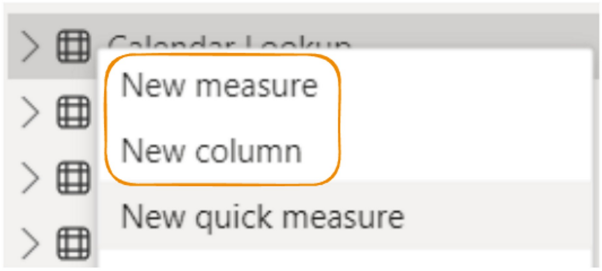
Report View:

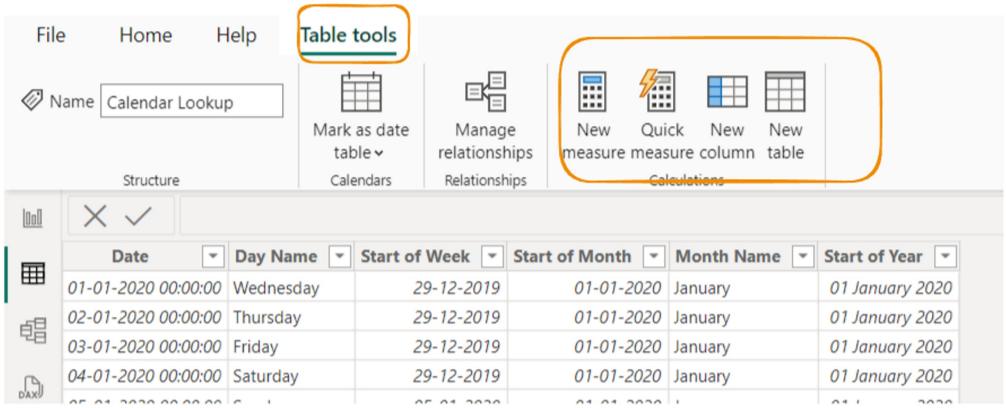
Category Name	Total Returns	Bike Returns
Accessories	1,115	
Bikes	427	427
Clothing	267	
Total	**1,809**	**427**

Explicit Measures

```
1 Bike Returns =
2 CALCULATE(
3     [Total Returns],          // Existing Measure
4     'Product Categories Lookup'[Category Name] = "Bikes"    // filtered for bikes only
5 )
```







Date	Day Name	Start of Week	Start of Month	Month Name	Start of Year
01-01-2020 00:00:00	Wednesday	29-12-2019	01-01-2020	January	01 January 2020
02-01-2020 00:00:00	Thursday	29-12-2019	01-01-2020	January	01 January 2020
03-01-2020 00:00:00	Friday	29-12-2019	01-01-2020	January	01 January 2020
04-01-2020 00:00:00	Saturday	29-12-2019	01-01-2020	January	01 January 2020

nn

e number

ure

```
IF(LogicalTest, ResultIfTrue, [ResultIfFalse])
```

Checks whether a condition is met, and returns one value if TRUE, and another value if FALSE.

```
1 Parent = IF(
```

```
Parent = IF('Customer Lookup'[TotalChildren] > 0,"Yes","No")
```

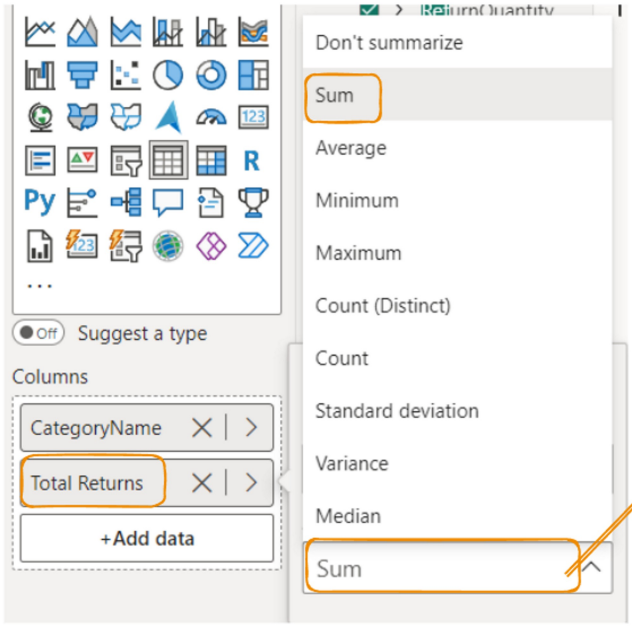
Parent

TotalChildren	EducationLevel	Occupation	HomeOwner	Domain Name	Parent
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No
0	Bachelors	Professional	N	Adventure Works	No

Relative Reference

TotalQuantity
84174
84174
84174
84174
84174
84174
84174
84174

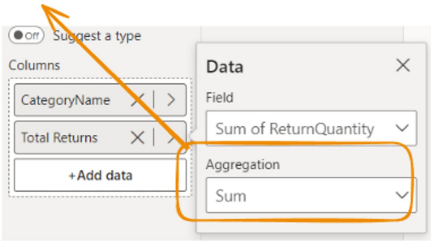
```
TotalQuantity = SUM('Sales Record 2020-2022'[OrderQuantity])
```



CategoryName	Total Returns
Accessories	1130
Bikes	429
Clothing	269
Total	**1828**

Implicit Measure -Report View

Rows
Columns
Value
Filter



values

CALCULATED COLUMNS

Calculated columns allow you to add new, formula-based columns to tables in a model

- Calculated columns refer to entire tables or columns (no A1-style cell references).

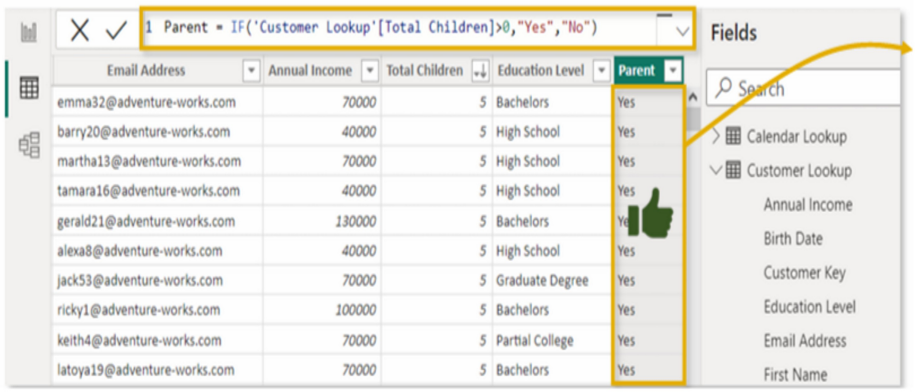
- Calculated columns generate values for each row, which are visible within tables in the Data view.

- Calculated columns understand row context; they're great for defining properties based on information in each row, but generally useless for aggregation (sum, count, etc.)

HEY THIS IS IMPORTANT!
As a rule of thumb, use calculated columns to "stamp" static, fixed values to each row in a table (or go upstream and use the Query Editor!)
DO NOT use calculated columns for aggregation – this is what measures are for!

PRO TIP:
Calculated columns are typically used for filtering & grouping data, rather than creating aggregate numerical values

EXAMPLE: CALCULATED COLUMNS

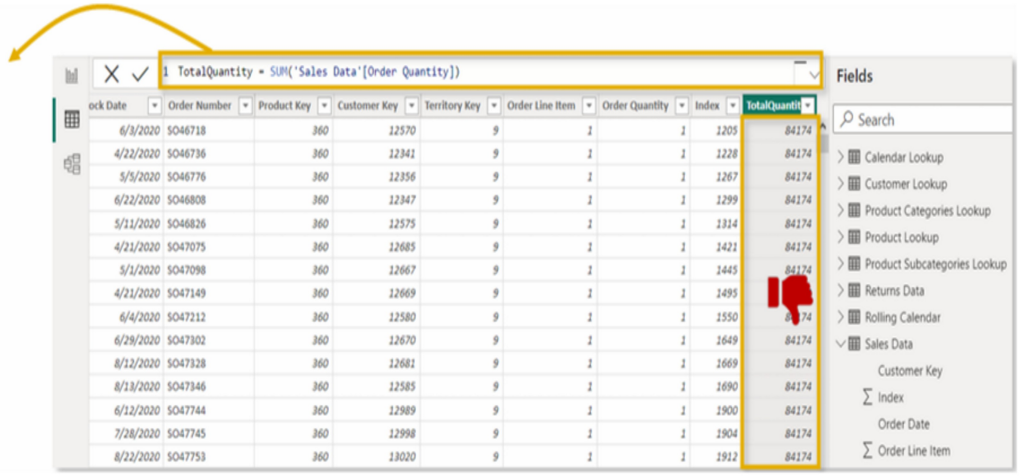


In this case we've added a calculated column named Parent, which equals "Yes" if the [Total Children] field is greater than 0, and "No" otherwise

- Since calculated columns understand row context, a new value is calculated in each row based on the value in the [Total Children] column.

- This is a valid use of calculated columns; it creates a new row "property" that we can use to filter or segment any related data within the model.

Here we're using an aggregation function (SUM) to calculate a new column named Total Quantity



- Since this is an aggregation function, the same grand total is returned in every row of the table

- This is not a valid use of calculated columns; these values are statically "stamped" onto the table and can't be filtered, sliced, etc.

DAX MEASURES

Measures are DAX formulas used to generate new calculated values

- Like calculated columns, measures reference entire tables or columns (no A1-style cell references).

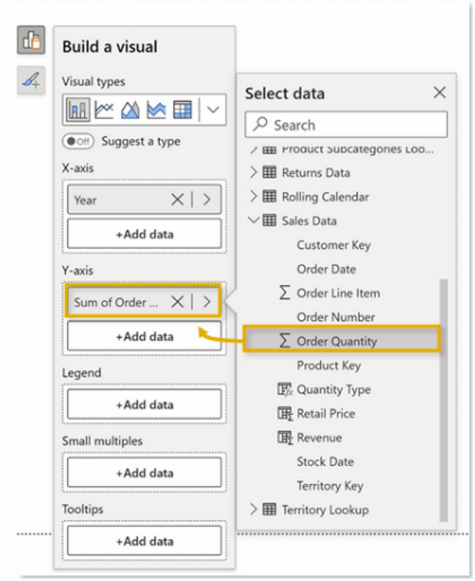
- Unlike calculated columns, measures aren't visible within tables; they can only be "seen" within a visualization like a chart or matrix (similar to a calculated field in a PivotTable).

- Measures evaluate based on filter context, which means they recalculate when the fields or filters around them change.

HEY THIS IS IMPORTANT!
As a rule of thumb, use measures when a single row can't give you the answer, or when you need to aggregate values across multiple rows in a table

PRO TIP:
Use measures to create numerical, calculated values that can be analyzed in the "values" field of a report visual

IMPLICIT VS. EXPLICIT MEASURES



Example of an implicit measure

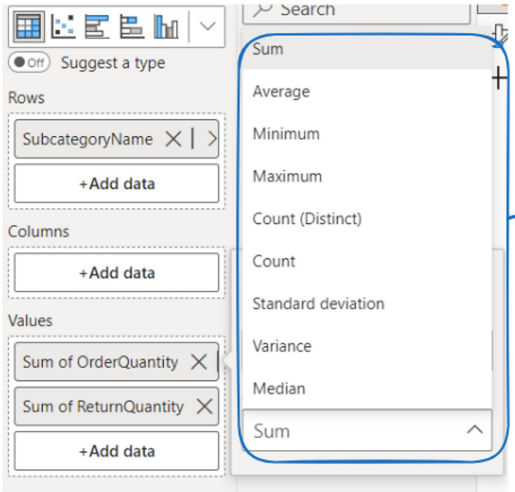
Implicit measures are created when you drag raw numerical fields into a report visual and manually select an aggregation mode (Sum, Average, Min, Max, Count, etc.)

Explicit measures are created when you actually write a DAX formula and define a new measure that can be used within the model

HEY THIS IS IMPORTANT!

Implicit measures are only accessible within the specific visualization in which they were created, and cannot be referenced elsewhere.

Explicit measures can be used anywhere in the report, and referenced by other DAX calculations to create "measure trees".



Implicit Measures.

Explicit Measure

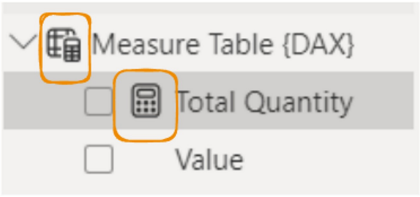
Black_Quantity_Sold = 10,590

Europe

Pacific

North America

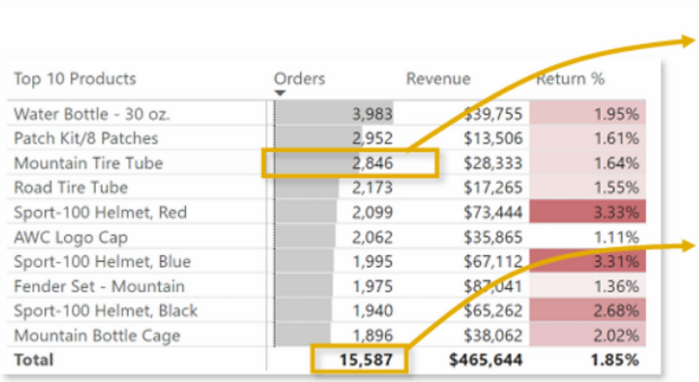
Europe_QS-black




```
Total Quantity = SUM('Sales Record 2020-2022'[OrderQuantity])
```

FILTER CONTEXT

Measures are evaluated based on filter context, which means that they recalculate whenever the fields or filters around them change



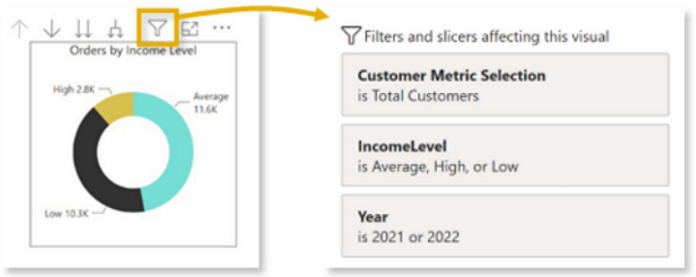
Top 10 Products	Orders	Revenue	Return %
Water Bottle - 30 oz.	3,983	$39,755	1.95%
Patch Kit/8 Patches	2,952	$13,506	1.61%
Mountain Tire Tube	2,846	$28,333	1.64%
Road Tire Tube	2,173	$17,265	1.55%
Sport-100 Helmet, Red	2,099	$73,444	3.33%
AWC Logo Cap	2,062	$35,865	1.11%
Sport-100 Helmet, Blue	1,995	$67,112	3.31%
Fender Set - Mountain	1,975	$87,041	1.36%
Sport-100 Helmet, Black	1,940	$65,262	2.68%
Mountain Bottle Cage	1,896	$38,062	2.02%
Total	**15,587**	**$465,644**	**1.85%**

For this value in the matrix (2,846), the Orders measure is calculated based on the following filter context: Products[Product Name] = "Mountain Tire Tube"

- This allows the measure to return the total order quantity for each product specifically (or whatever context the row and column labels dictate – years, countries, categories, customer names, etc.)

This total (15,587) does NOT calculate by summing the values above; it evaluates as an independent measure with no filter context applied

- IMPORTANT: Every measure value in a report evaluates independently (like an island) and calculates based on its own filter context



PRO TIP: Clicking the filter icon will show you the filters currently applied to a selected visual

EXAMPLE: FILTER CONTEXT

MEASURE: Revenue Per Customer

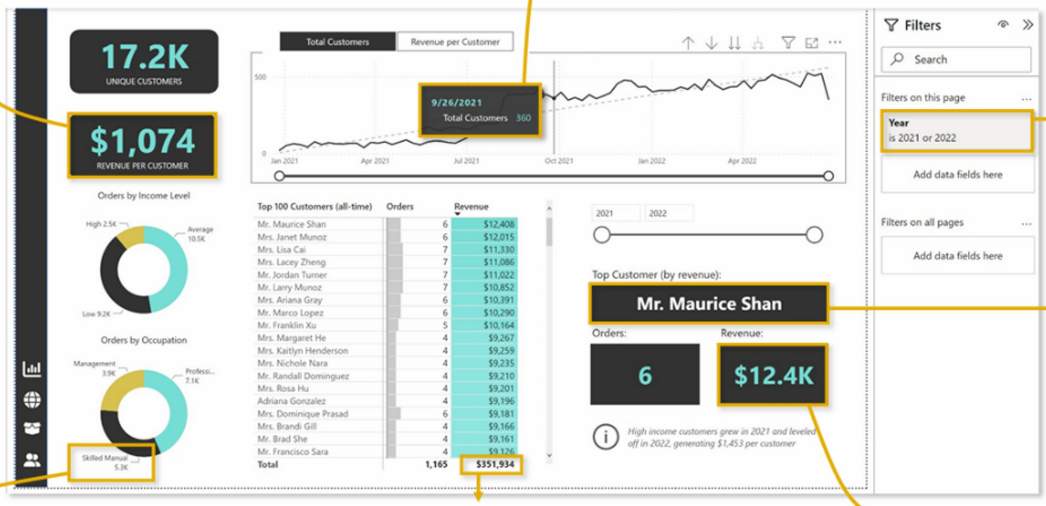
FILTER CONTEXT:

- *Calendar[**Year**]* = 2021 *or* 2022

MEASURE: Total Customers

FILTER CONTEXT:

- *Calendar[**Date**]* = September 26, 2021



This is **a page-level filter**, which impacts **ALL** visuals on this report page *(more on this later!)*

COLUMN: Customer Full Name

FILTER CONTEXT:

- *Calendar[**Year**]* = 2021 *or* 2022
- *Customer[**Full Name**]* = Top 1 by Total Revenue

MEASURE: Total Orders

FILTER CONTEXT:

- *Calendar[**Year**]* = 2021 *or* 2022
- *Customers[**Occupation**]* = Skilled Manual

MEASURE: Total Revenue

FILTER CONTEXT:

- *Calendar[**Year**]* = 2021 *or* 2022
- *Customer[**Full Name**]* = Top 100 by Total Orders

MEASURE: Total Revenue

FILTER CONTEXT:

- *Calendar[**Year**]* = 2021 *or* 2022
- *Customer[**Full Name**]* = Mr. Maurice Shan

STEP-BY-STEP MEASURE CALCULATION

Product Color	Quantity Sold
Black	10,590
Red	4,011
Yellow	4,638

How exactly is this measure value calculated?

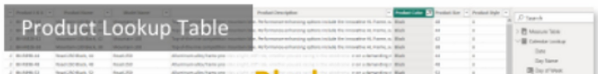
- NOTE: This all happens instantly behind the scenes, every time the filter context changes

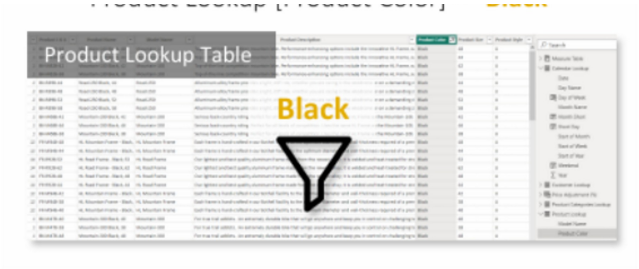
STEP 1

Filter context is detected & applied

Product Color	Quantity Sold
Black	10,590
Red	4,011
Yellow	4,638

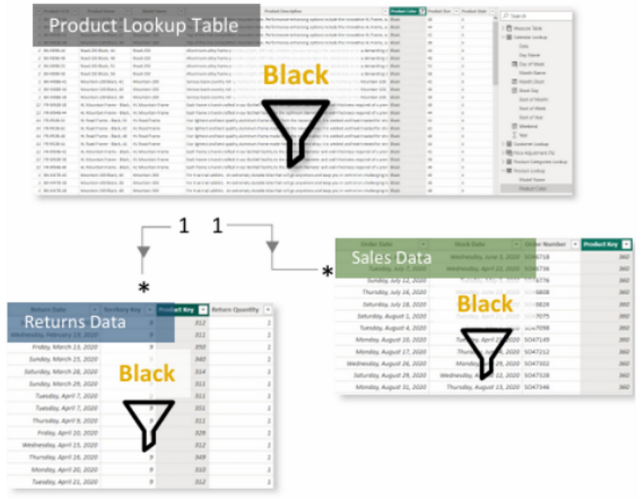
'Product Lookup'[Product Color] = "Black"





STEP 2

Filters flow “downstream” to related tables



STEP 3

Measure evaluates against the filtered table

```
1  Quantity Sold =
2  SUM(
3  |    'Sales Data'[Order Quantity]
4  )
```

*Sum of values in the **Order Quantity** column of the **Sales Data** table, filtered to rows where the product color is “**Black**”*

= 10,590

DAX SYNTAX

MEASURE NAME

- Measures are always surrounded by 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

This is a "fully qualified" column, since it's preceded by the table name.

NOTE: Table names with spaces must be surrounded by single quotes:

- Without a space: Transactions[quantity]
- With a space: 'Transactions Table'[quantity]

PRO TIP:
Column references use fully qualified names (i.e. 'Table'[Column])

Measure references just use the measure name (i.e. [Measure]) and can be called by typing an open square bracket " ["