CALCULATED FIELD WITH DAX

Meet Dax

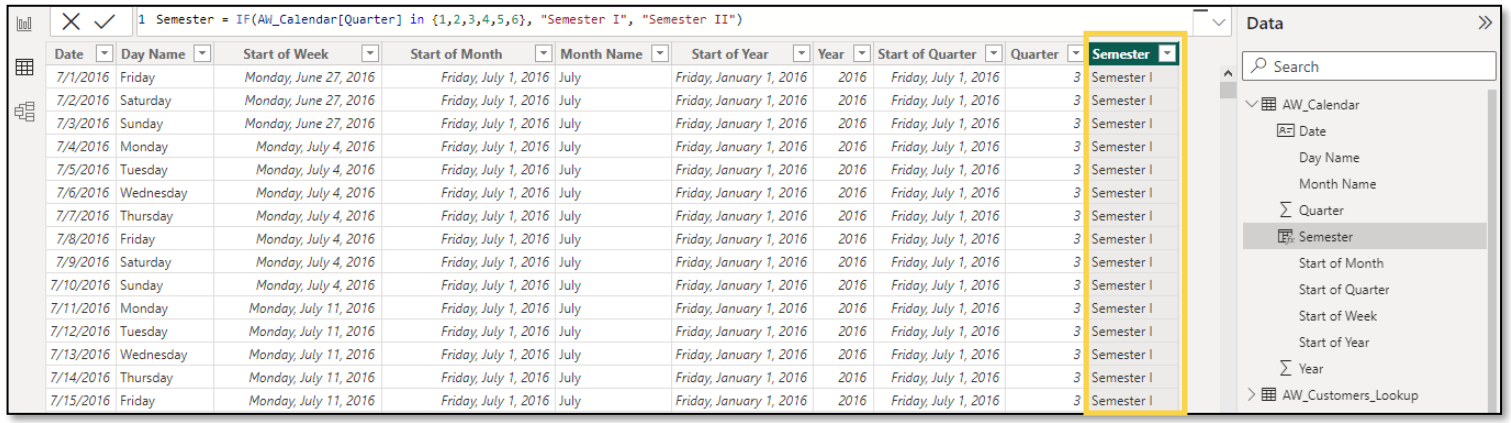
Data Analysis Expressions, commonly known as **DAX**, is the formula language that drives Power BI. With DAX, you can:

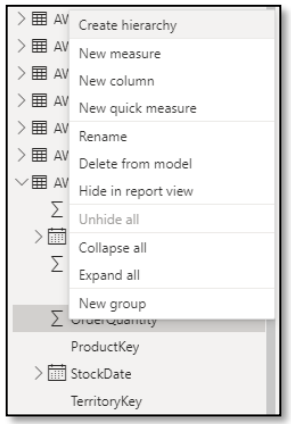
- Add **calculated columns** and **measures** to your model, using intuitive syntax

Two ways to use Dax

1) Calculated Columns

2) New Measure





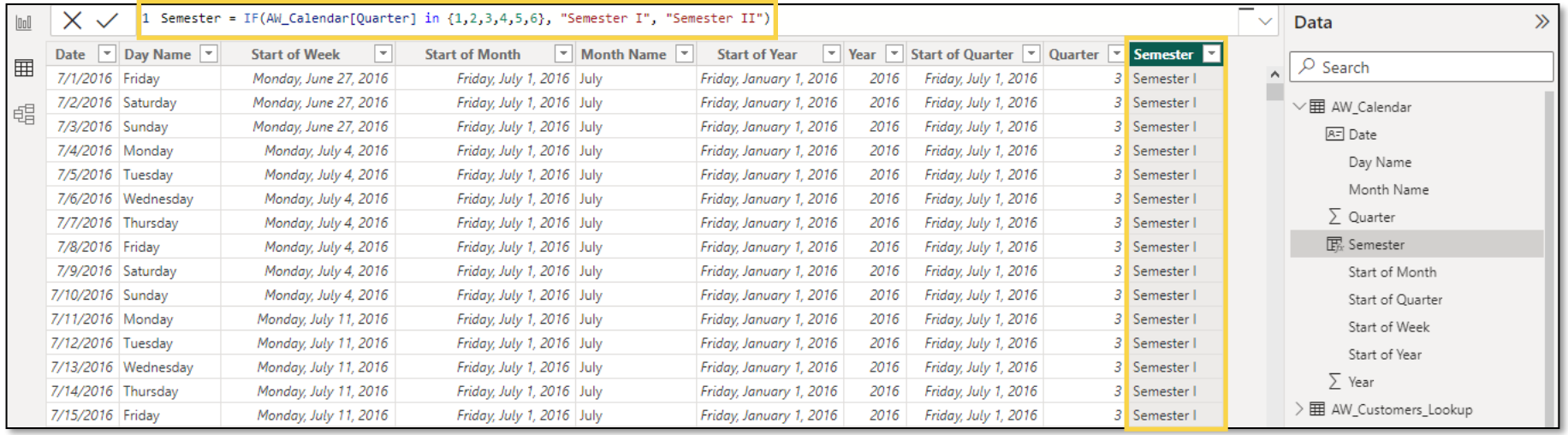
Calculated Column

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

- Calculated columns refer to **entire tables** or columns
- Calculated columns generate values for each row, which are **visible within tables in the Data view**
- Recalculate on data source refresh or when changes are made to component columns
- Primarily used as **rows**, **columns**, **slicers** or **filters**

PRO TIP:
DO NOT use calculated columns for aggregation formulas, or to calculate fields for the “Values” area of a visualization (use measures instead)

Calculated Column (Example)



Date	Day Name	Start of Week	Start of Month	Month Name	Start of Year	Year	Start of Quarter	Quarter	Semester
7/1/2016	Friday	Monday, June 27, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/2/2016	Saturday	Monday, June 27, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/3/2016	Sunday	Monday, June 27, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/4/2016	Monday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/5/2016	Tuesday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/6/2016	Wednesday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/7/2016	Thursday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/8/2016	Friday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/9/2016	Saturday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/10/2016	Sunday	Monday, July 4, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/11/2016	Monday	Monday, July 11, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/12/2016	Tuesday	Monday, July 11, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/13/2016	Wednesday	Monday, July 11, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/14/2016	Thursday	Monday, July 11, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I
7/15/2016	Friday	Monday, July 11, 2016	Friday, July 1, 2016	July	Friday, January 1, 2016	2016	Friday, July 1, 2016	3	Semester I

In this case we've added a **calculated column** named **"Semester"**, which equals **"Semester I"** if the (AW_Calendar[Quarter]) field is 1,2,3,4,5,6 and **"Semester II"** otherwise.

Measures

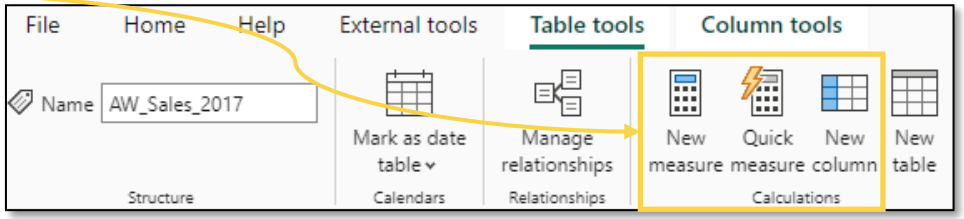
Measures are DAX formulas used to generate new calculated values

- Like calculated columns, **measures** reference **entire tables** or **columns**
- Unlike calculated columns, **measure** values aren't visible within tables; they can only be "seen" within a visualization like a chart or matrix.
- Recalculate in response to any change to filters within the reports
- Almost always used within the **values** field of a visual

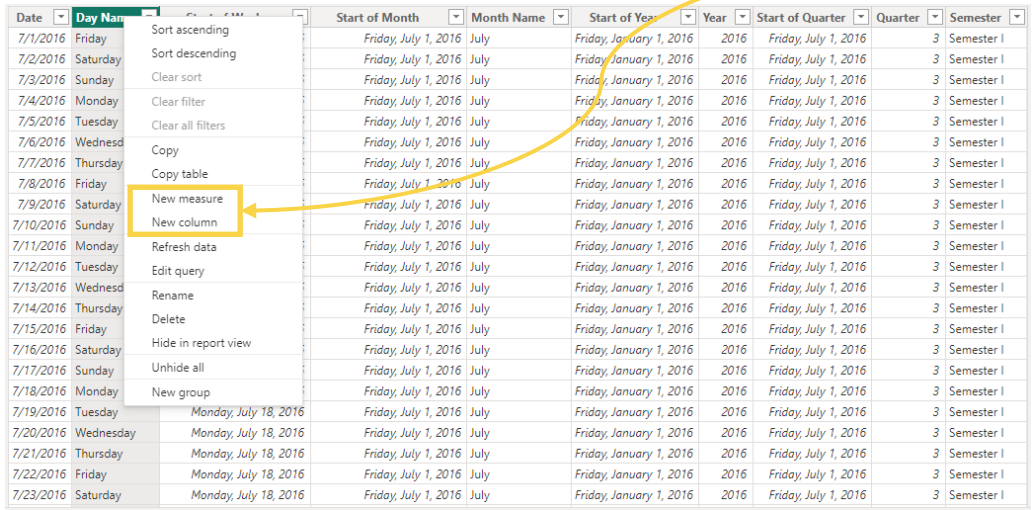
PRO TIP:
Use measures to create **numerical, calculated** values that can be analyzed in the "**values**" field of a report visual

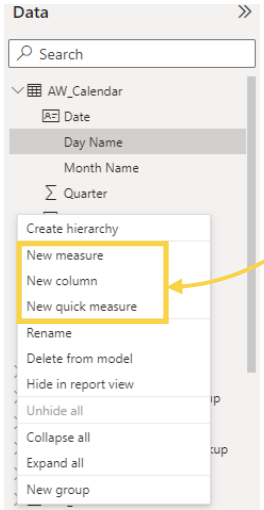
Add Columns & Measures

Option 1: Select **"New Measure"** or **"New Column"** from the **Table tools** tab

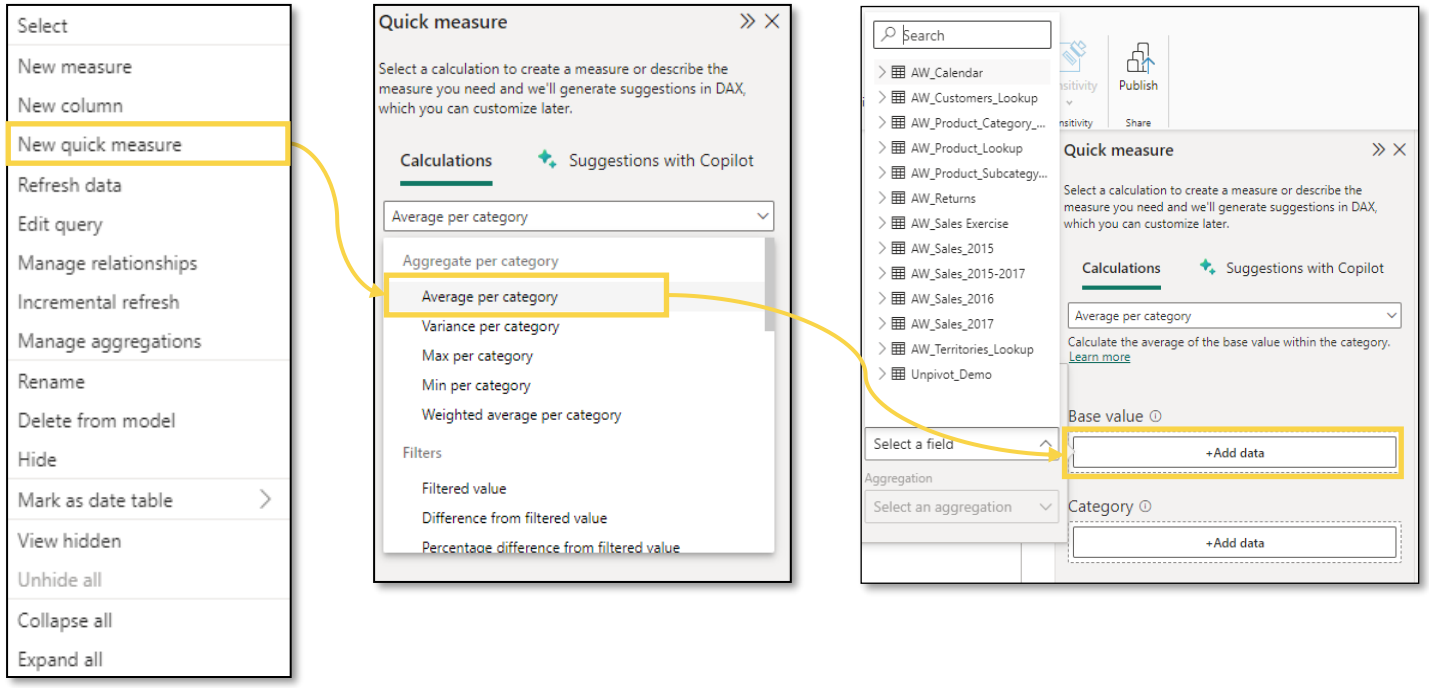


Option 2: Right-click within the table (in the **Data view**) or the Field List (in either the **Data** or **Report view**)





Quick Measures



Quick Measures are pre-built formula templates that allow you to drag and drop fields, rather than write DAX from scratch

Understanding Filter Context

ProductName	Total Order	Total Return
Water Bottle - 30 oz.	4,565	155
Patch Kit/8 Patches	3,255	95
Mountain Tire Tube	3,121	93
Road Tire Tube	2,379	67
Fender Set - Mountain	2,222	54
Mountain Bottle Cage	2,190	77
Road Bottle Cage	1,943	56
Touring Tire Tube	1,572	45
Sport-100 Helmet, Red	1,218	70
Sport-100 Helmet, Blue	1,152	66
ML Mountain Tire	1,132	28
Sport-100 Helmet, Black	1,074	52
LL Road Tire	1,069	43
Touring Tire	1,021	21
ML Road Tire	956	26
Bike Wash - Dissolver	949	25
LL Mountain Tire	856	39
HL Mountain Tire	736	49
HL Road Tire	406	28
Hydration Pack - 70 oz.	394	25
Hitch Rack - 4-Bike	167	8
All-Purpose Bike Stand	116	8
Total	**32,493**	**1,130**

For this particular value in the matrix, the **Total Orders** measure is calculated based on the following filter context: *Products[**ProductName**]* = "Touring Tire Tube"

This Total is **not** calculated by summing the values above; it evaluates as its own measure, with **no filter context**

Filter Context (Examples)

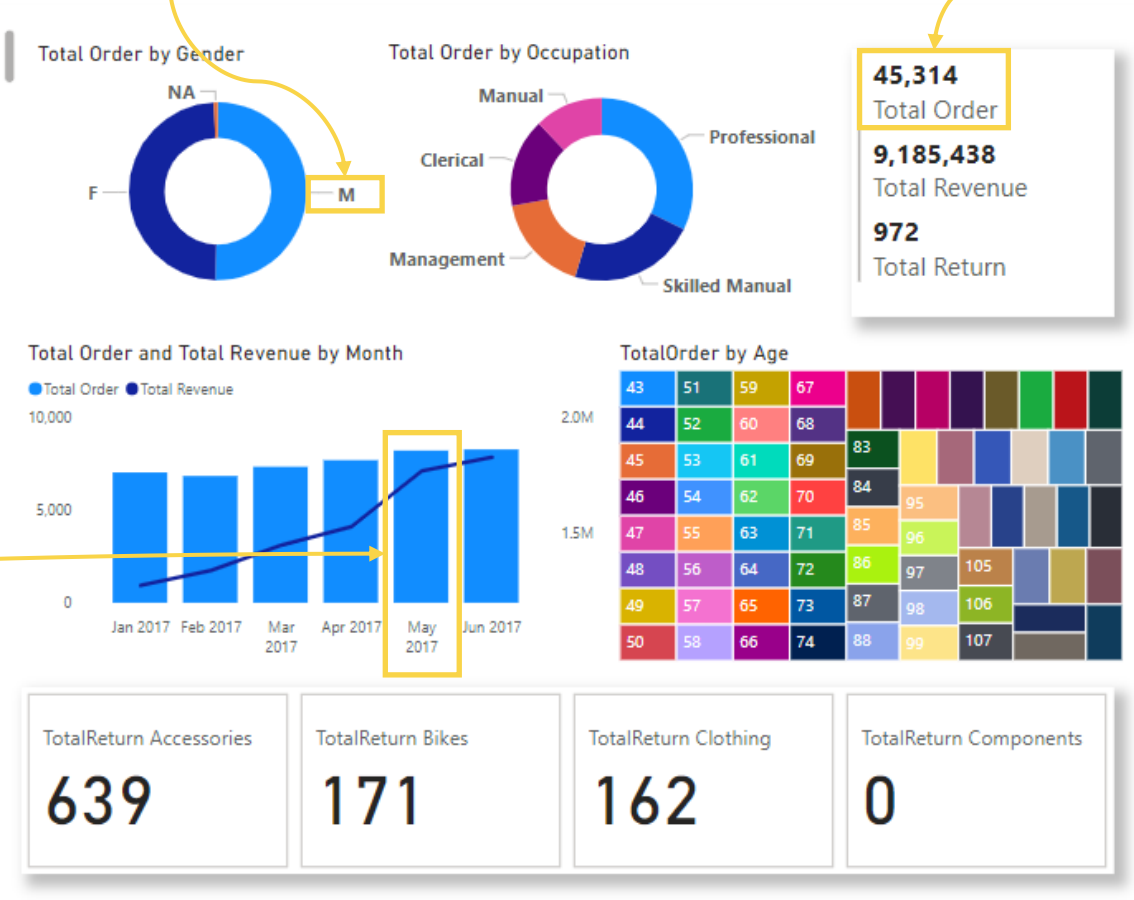
MEASURE: Total Revenue
Filter Context:
- *Calendar[Year]* = "2017"
- *Customer[Full Name]* = "Mr. Ryan Thompson"

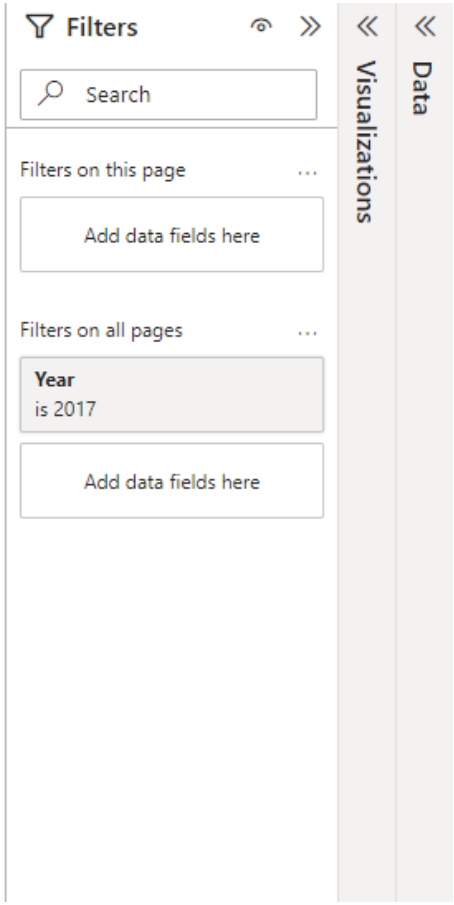
MEASURE: Total Order
Filter Context:
- *Calendar[Year]* = "2017"
- *Customer[Gender]* = "M (Male)"

MEASURE: Total Order
Filter Context:
- *Calendar[Year]* = "2017"

MEASURE: Total Order
Filter Context:
- *Calendar[Year]* = "2017"
- *Customer[Month]* = "May 2017"

FullName	Total Order	TotalRevenue
Mr. Fernando Barnes	74	1,250
Mrs. Jennifer Simmons	74	896
Mrs. Ashley Henderson	72	1,260
Mrs. Hailey Patterson	59	975
Mrs. Nancy Chapman	54	824
Mr. Ryan Thompson	53	872
Mr. Charles Jackson	52	944
Mr. Daniel Davis	48	852
Mr. Dalton Perez	46	845
Ms. April Shan	45	633
Mr. Henry Garcia	44	771
Mrs. Samantha Jenkins	40	655
Mrs. Samantha Russell	38	648
Mr. Jason Griffin	35	683
Mr. Eduardo Patterson	34	788
Mr. Mason Roberts	33	550
Ms. Sarah Simmons	32	630
Ms. Sierra Young	31	469
Mr. Jared Peterson	29	468
Mrs. Brandy Chandra	29	627
Mr. Luis Diaz	28	447
Mr. Luke Lal	28	562
Ms. Jasmine Powell	28	475
Mrs. Chloe Campbell	27	604
Mrs. Jada Morgan	24	434
Mr. Antonio Bennett	23	375
Mrs. Megan Jenkins	23	391
Mr. Nicholas Brown	22	444
Ms. Isabella Russell	22	449
Mr. Jonathan Chen	21	300
Mr. Marco Lopez	21	6,136
Mr. José Hernandez	20	401
Mr. Ricky Navarro	20	4,676
Total	**45,314**	**9,185,438**





Step by Step Measure Calculation

CategoryName	TotalReturn
Accessories	1,130
Bikes	429
Clothing	269
Total	**1,828**

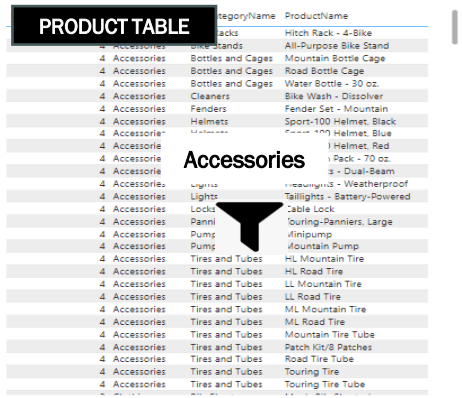
How exactly is this measure calculated?

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

STEP 1

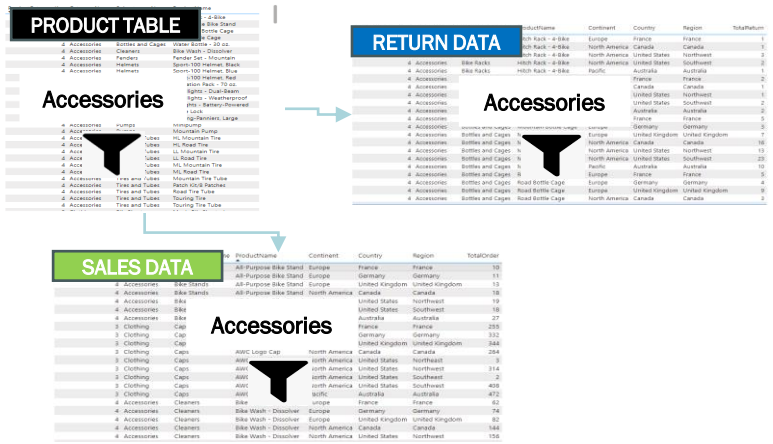
Filter context is detected & applied

Product[CategoryProductName]="Accessories"



STEP 2

Filters flow "downstream" to all related tables



STEP 3

Measure formula evaluates against the filtered table

```
1 TotalReturn = CALCULATE(sum(AW_Returns[ReturnQuantity]),
2 FILTER(AW_Product_Category_Lookup, AW_Product_Category_Lookup[CategoryName]="Accessories"))
```

Count of rows in the **AW_Returns**_Data table, filtered down to only rows where the product category is "Accessories"

= 1,130