DAX [Data Analysis Expression]

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.

Two ways to use DAX

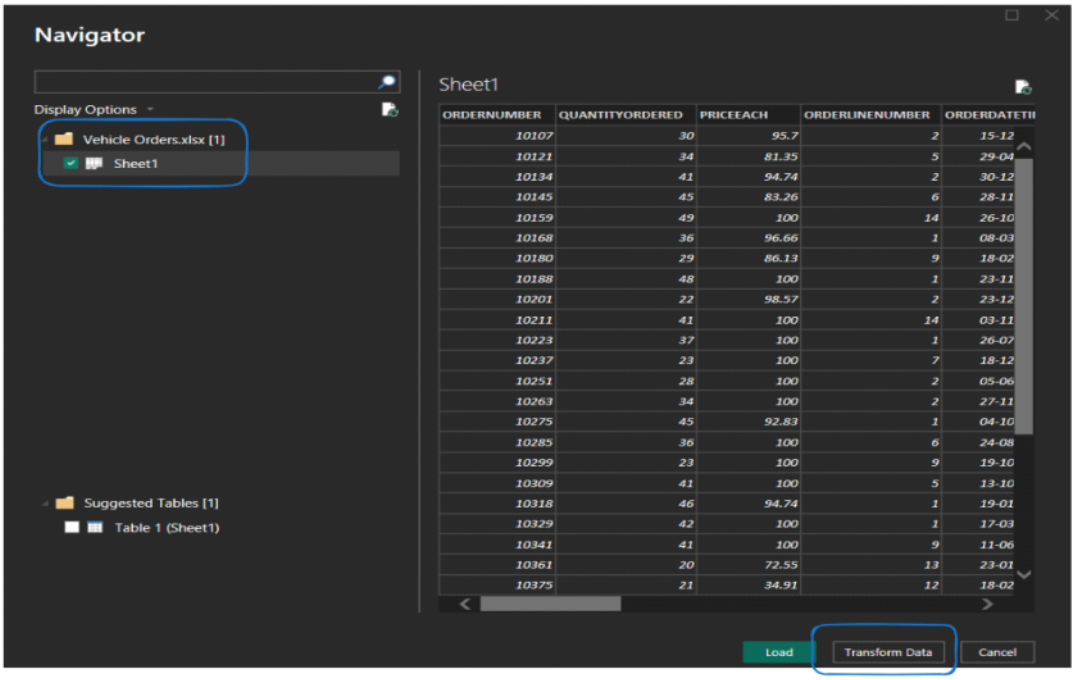
Calculated Columns
- Row Context.
- Filtering
- Data View

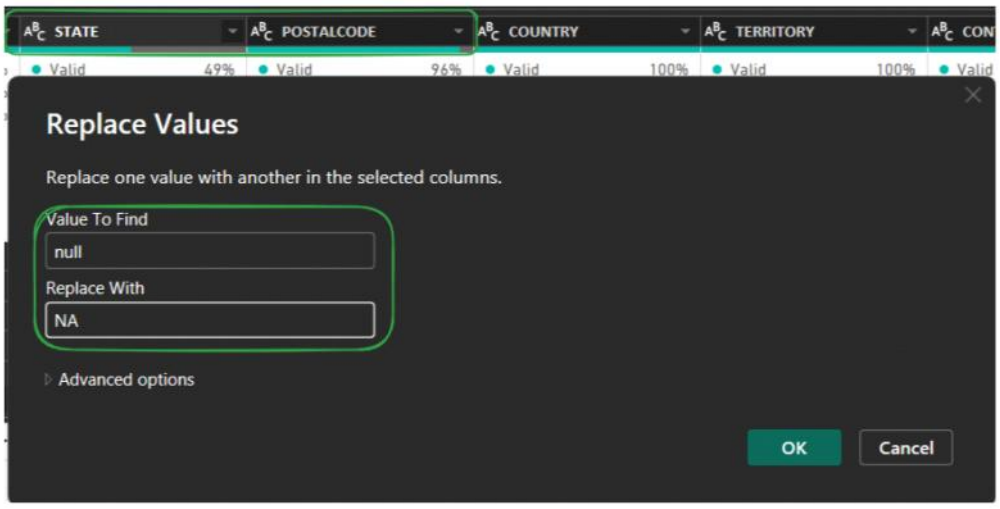
Measures
- Filter Context
- Aggregation.
- Report View

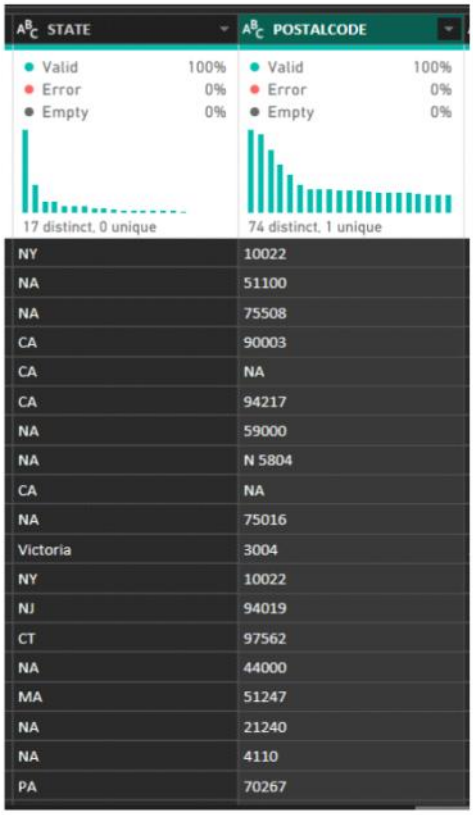
M VS. DAX

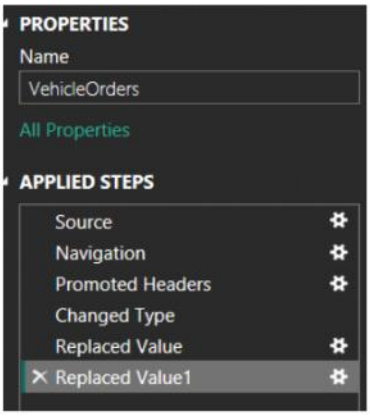
M and DAX are two distinct functional languages used within Power BI Desktop:

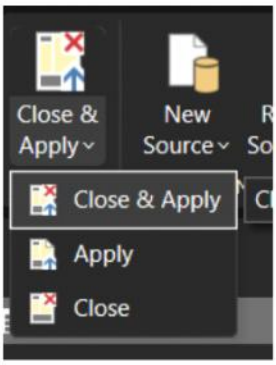
- M is used in the Power Query editor, and is designed specifically for extracting, transforming and loading data.
- DAX is used in the Power BI front-end, and is designed specifically for analyzing relational data models



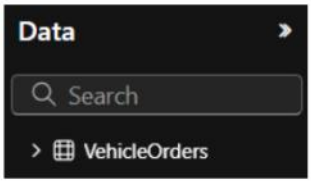

STATE
POSTALCODE
COUNTRY
TERRITORY
Valid 49%
Valid 96%
Valid 100%
Valid 100%
Replace Values
Replace one value with another in the selected columns.
Value To Find
null
Replace With
NA
Advanced options
OK
Cancel

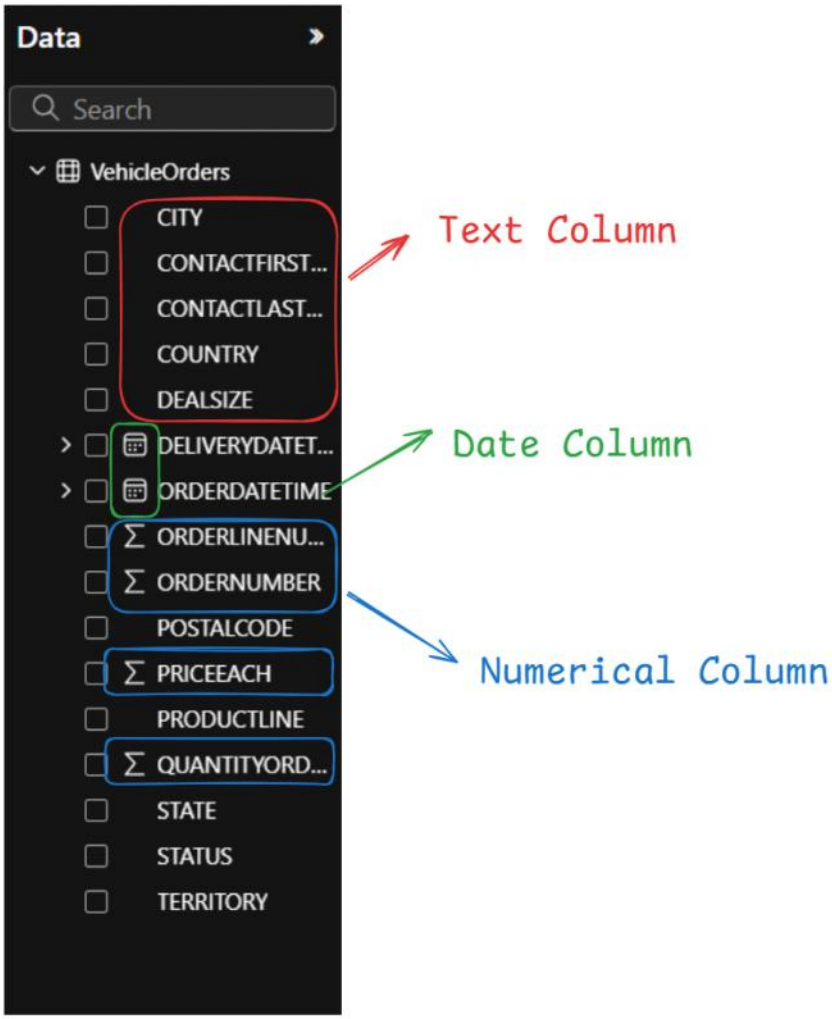

STATE
POSTALCODE
Valid 100%
Error 0%
Empty 0%
17 distinct, 0 unique
74 distinct, 1 unique
NY 10022
NA 51100
NA 75508
CA 90003
CA NA
CA 94217
NA 59000
NA N 5804
CA NA
NA 75016
Victoria 3004
NY 10022
NJ 94019
CT 97562
NA 44000
MA 51247
NA 21240
NA 4110
PA 70267


PROPERTIES
Name
VehicleOrders
All Properties
APPLIED STEPS
Source
Navigation
Promoted Headers
Changed Type
Replaced Value
Replaced Value1

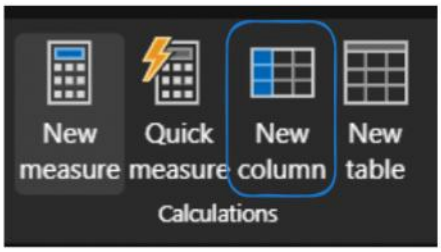

Close & Apply
New Source
Close & Apply
Apply
Close


Load
VehicleOrders
Creating connection in model...
Cancel


Data
Search
VehicleOrders



ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	ORDERDATETIME	DELIVERYDATETIME	STATUS	PRODUCTLINE
10203	34	$64.90	7	09-09-2013 00:20:00	11-09-2013 23:06:10	Shipped	Classic Cars
10203	21	$37.00	2	09-11-2013 00:31:00	11-11-2013 23:57:44	Shipped	Classic Cars
10203	33	$86.04	11	11-07-2013 23:27:00	13-07-2013 20:33:06	Shipped	Classic Cars
10203	45	$85.47	4	02-02-2013 01:50:00	02-02-2013 22:07:58	Shipped	Classic Cars
10203	44	$82.99	9	02-04-2016 23:05:00	04-04-2016 06:37:04	Shipped	Classic Cars
10203	48	$100.00	1	05-07-2013 23:20:00	08-07-2013 20:59:14	Shipped	Classic Cars
10203	20	$100.00	6	03-04-2012 23:48:00	05-04-2012 10:25:30	Shipped	Classic Cars
10203	47	$100.00	3	07-01-2013 00:48:00	08-01-2013 12:22:10	Shipped	Classic Cars
10203	32	$100.00	10	03-04-2015 00:10:00	04-04-2015 12:16:26	Shipped	Classic Cars
10203	47	$100.00	5	12-03-2014 00:54:00	13-03-2014 17:57:38	Shipped	Classic Cars
10203	20	$100.00	8	19-05-2014 23:07:00	21-05-2014 11:19:59	Shipped	Classic Cars



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

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).

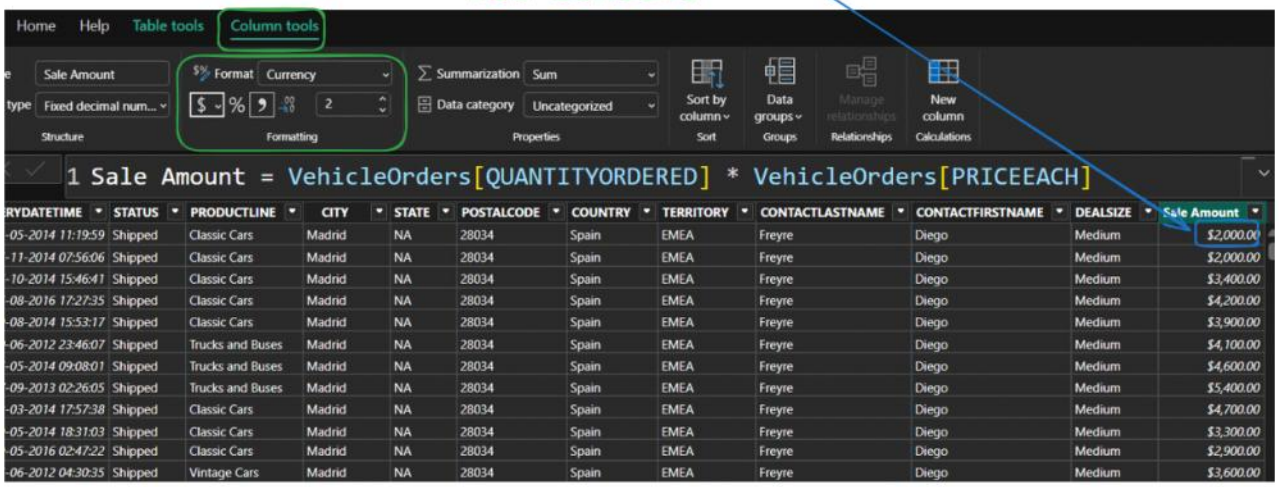
- Measures evaluate based on filter context, which means they recalculate when the fields or filters around them change.

Filter Context

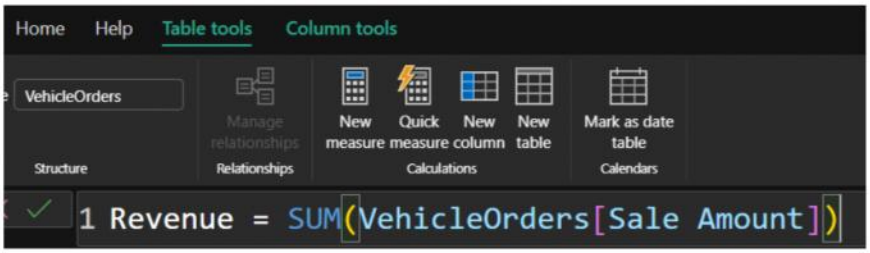
```
1 Sale Amount = VehicleOrders[QUANTITYORDERED] * VehicleOrders[PRICEEACH]
```

ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	ORDERDATETIME	DELIVERYDATETIME	STATUS	PRODUCTLINE	CITY	STATE	POSTALCODE
10203	20	$100.00	8	19-05-2014 23:07:00	21-05-2014 11:19:59	Shipped	Classic Cars	Madrid	NA	28034
10153	20	$100.00	11	25-11-2014 01:04:00	25-11-2014 07:56:06	Shipped	Classic Cars	Madrid	NA	28034
10104	34	$100.00	1	13-10-2014 00:33:00	14-10-2014 15:46:41	Shipped	Classic Cars	Madrid	NA	28034
10153	42	$100.00	12	14-08-2016 23:02:00	16-08-2016 17:27:35	Shipped	Classic Cars	Madrid	NA	28034
10212	39	$100.00	16	28-08-2014 00:16:00	29-08-2014 15:53:17	Shipped	Classic Cars	Madrid	NA	28034
10104	41	$100.00	9	17-06-2012 23:51:00	20-06-2012 23:46:07	Shipped	Trucks and Buses	Madrid	NA	28034
10246	46	$100.00	5	15-05-2014 23:47:00	16-05-2014 09:08:01	Shipped	Trucks and Buses	Madrid	NA	28034
10412	54	$100.00	5	26-09-2013 23:42:00	27-09-2013 02:26:05	Shipped	Trucks and Buses	Madrid	NA	28034
10203	47	$100.00	5	12-03-2014 00:54:00	13-03-2014 17:57:38	Shipped	Classic Cars	Madrid	NA	28034
10212	33	$100.00	15	07-05-2014 01:19:00	09-05-2014 18:31:03	Shipped	Classic Cars	Madrid	NA	28034
10212	29	$100.00	10	09-05-2016 00:47:00	09-05-2016 02:47:22	Shipped	Classic Cars	Madrid	NA	28034

Row-Context



RYDATETIME	STATUS	PRODUCTLINE	CITY	STATE	POSTALCODE	COUNTRY	TERRITORY	CONTACTLASTNAME	CONTACTFIRSTNAME	DEALSIZE	Sale Amount
-05-2014 11:19:59	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$2,000.00
-11-2014 07:56:06	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$2,000.00
-10-2014 15:46:41	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$3,400.00
-08-2016 17:27:35	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$4,200.00
-08-2014 15:53:17	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$3,900.00
-06-2012 23:46:07	Shipped	Trucks and Buses	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$4,100.00
-05-2014 09:08:01	Shipped	Trucks and Buses	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$4,600.00
-09-2013 02:26:05	Shipped	Trucks and Buses	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$5,400.00
-03-2014 17:57:38	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$4,700.00
-05-2014 18:31:03	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$3,300.00
-05-2016 02:47:22	Shipped	Classic Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$2,900.00
-06-2012 04:30:35	Shipped	Vintage Cars	Madrid	NA	28034	Spain	EMEA	Freyre	Diego	Medium	$3,600.00

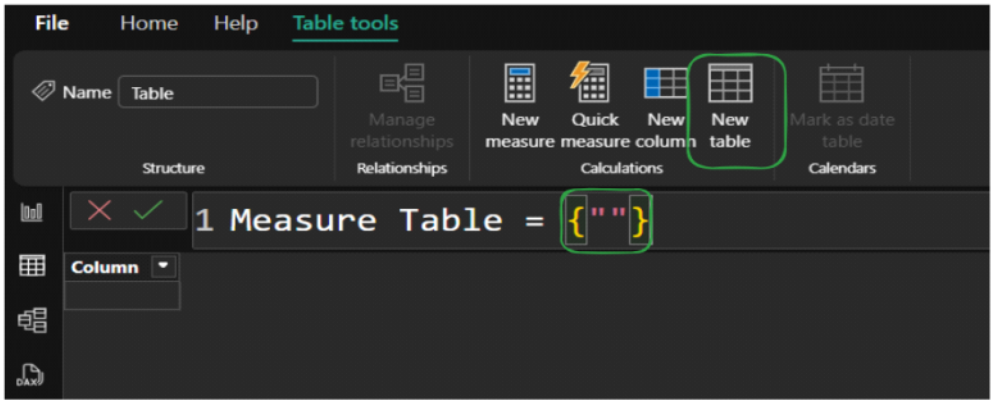


Revenue
$82,90,886.79
$82,90,886.79
$82,90,886.79
$82,90,886.79
$82,90,886.79
$82,90,886.79
$82,90,886.79
$82,90,886.79

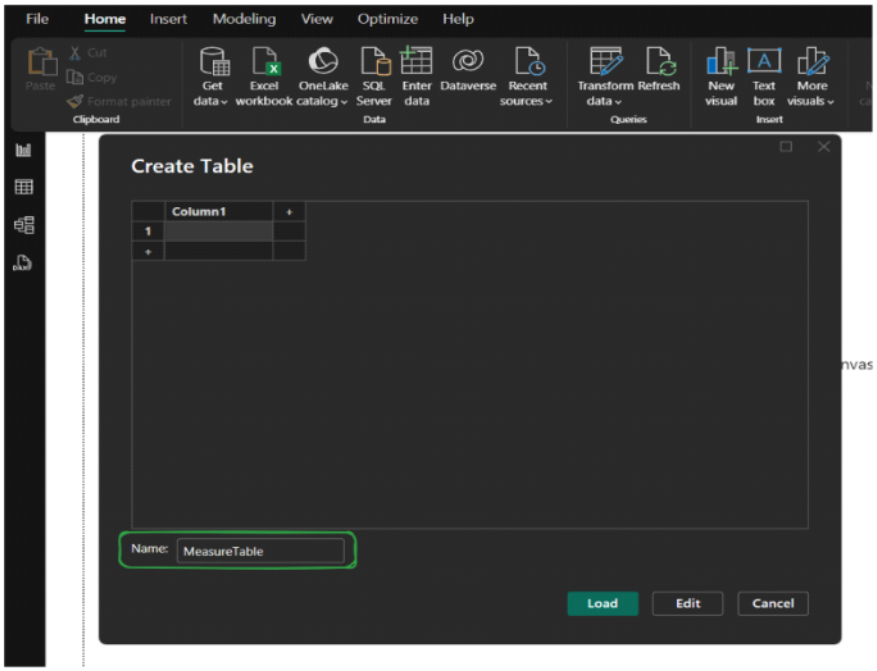
Aggregation in Calculated Column is Useless ❌

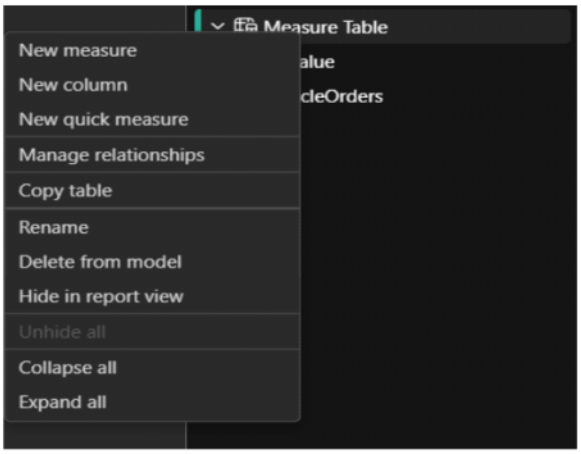
Measure Table

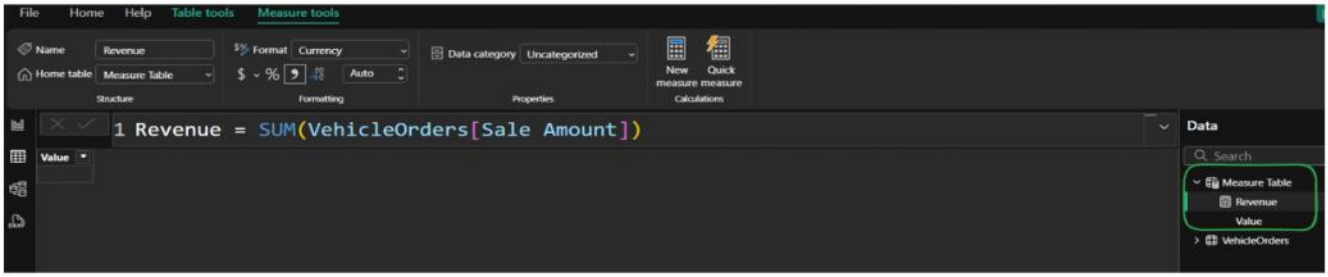
Option1:

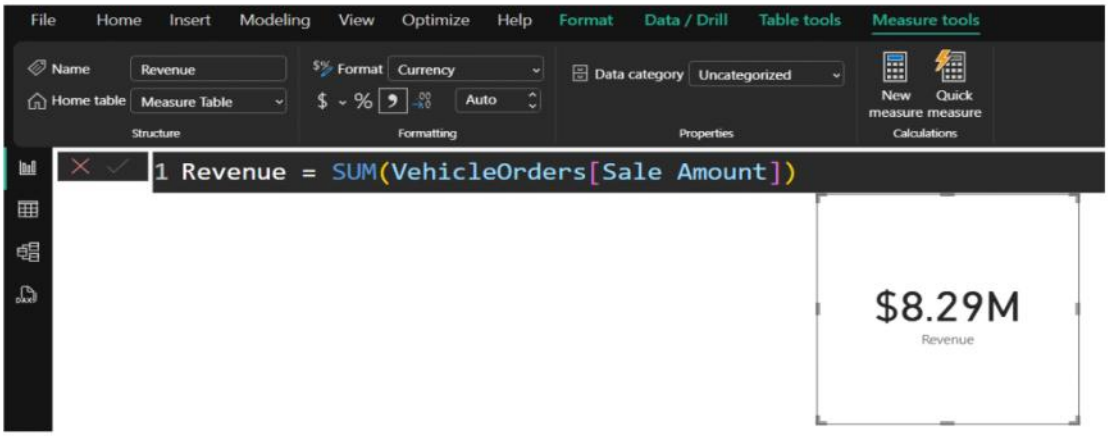


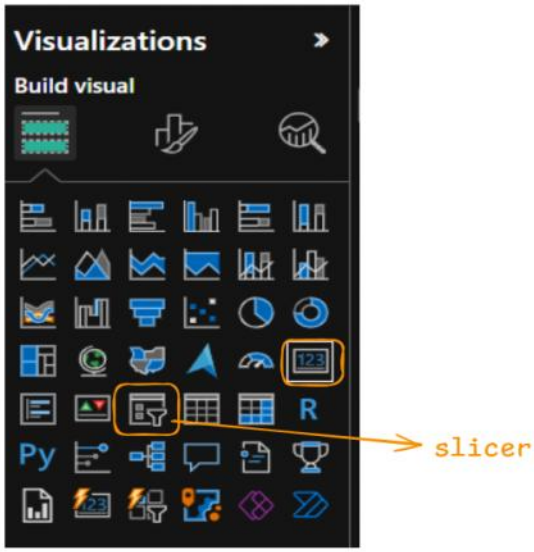
Option2:











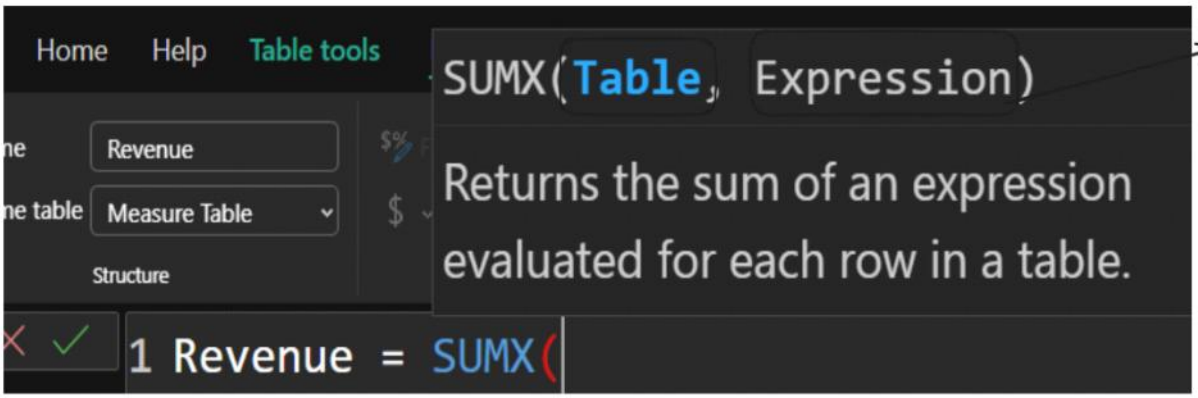
SUMX - Iterative Functions

QUANTITYORDERED	PRICEEACH
20	$100.00
20	$100.00
34	$100.00
42	$100.00
39	$100.00
41	$100.00
46	$100.00
54	$100.00
47	$100.00
33	$100.00
29	$100.00

```
(qty1 * price1) + (qty2 * price2) + (qty3 * price3)+
.............. + (qtyN * priceN)
```

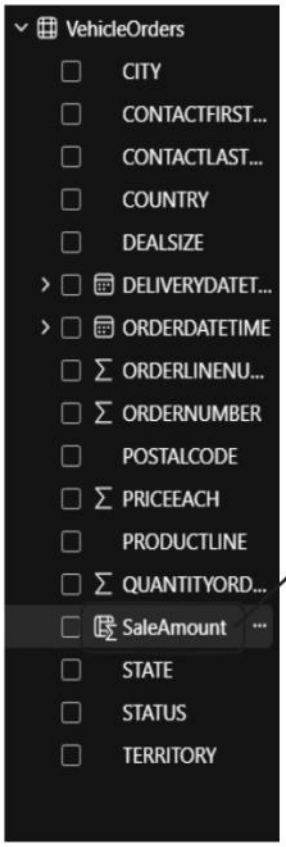
```
1 Revenue = SUM(VehicleOrders[Sale Amount])
```

Column 'Sale Amount' in table 'VehicleOrders' cannot be found or may not be used in this expression.

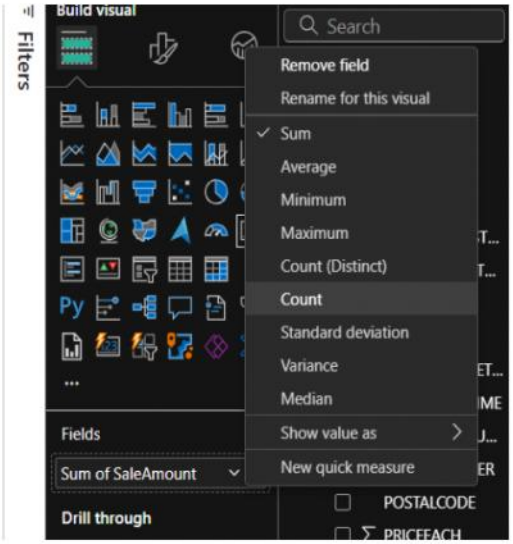


qty * price

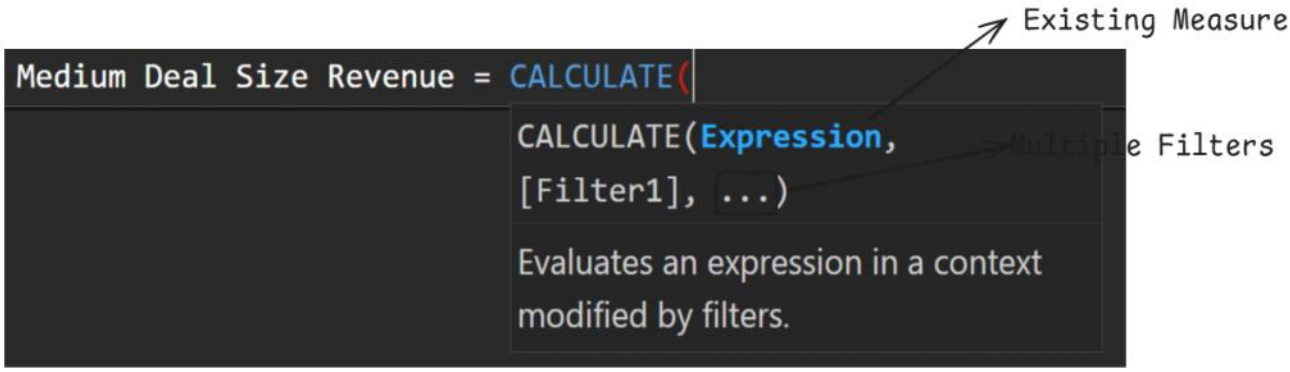
```
Revenue = SUMX(VehicleOrders,
            VehicleOrders[QUANTITYORDERED] * VehicleOrders[PRICEEACH]
        )
```

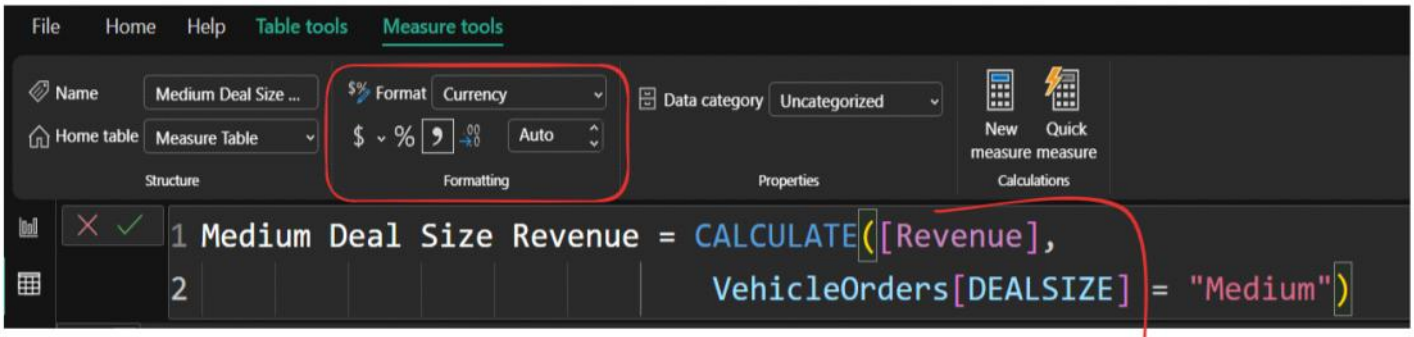


Calculated Column -> Drag into Report View
[SUM] Aggregator

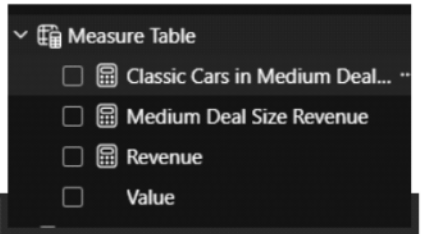


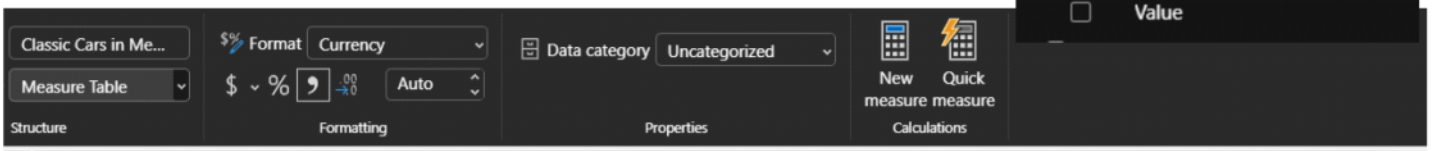
Note: If you can use the existing measure for further complex calculation : Explicit Measure





Explicit Measure






```
Classic Cars in Medium Deal Size Revenue = CALCULATE(
                                    [Medium Deal Size Revenue],
                                    VehicleOrders[PRODUCTLINE] = "Classic
                                    Cars")
```

IMPLICIT VS. EXPLICIT MEASURES

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".

RECAP: CALCULATED COLUMNS VS. MEASURES

CALCULATED COLUMNS	MEASURES
• Values are calculated based on information from each row of a table (row context)	• Values are calculated based on information from any filters in the report (filter context)
• Appends static values to each row in a table and stores them in the model (which increases file size)	• Does not create new data in the tables themselves (doesn't increase file size)
• Recalculate on data source refresh or when changes are made to component columns	• Recalculate in response to any change to filters within the report
• Primarily used for filtering data in reports	• Primarily used for aggregating values in report visuals