Microsoft

Model data with Power BI Desktop

aka.ms/PL300-3

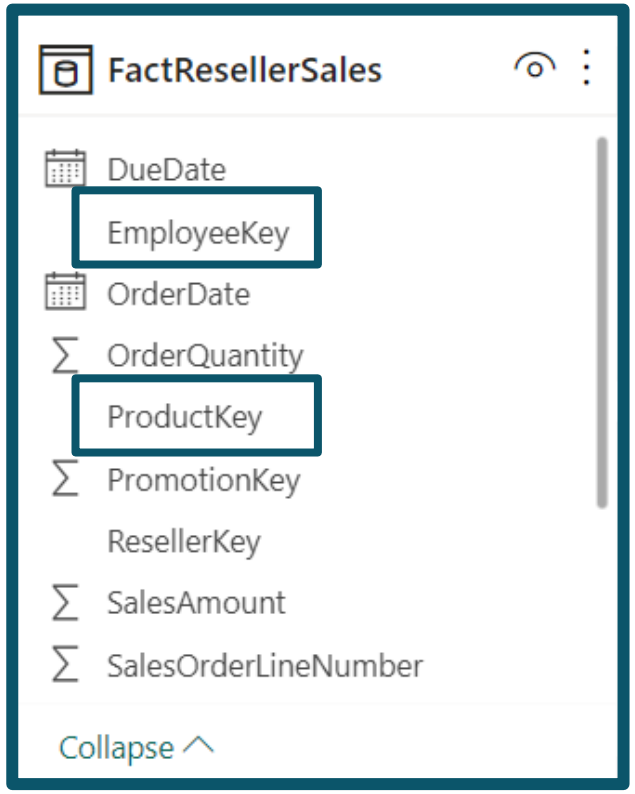
Agenda

- Introduction to data modeling
- Manage relationships

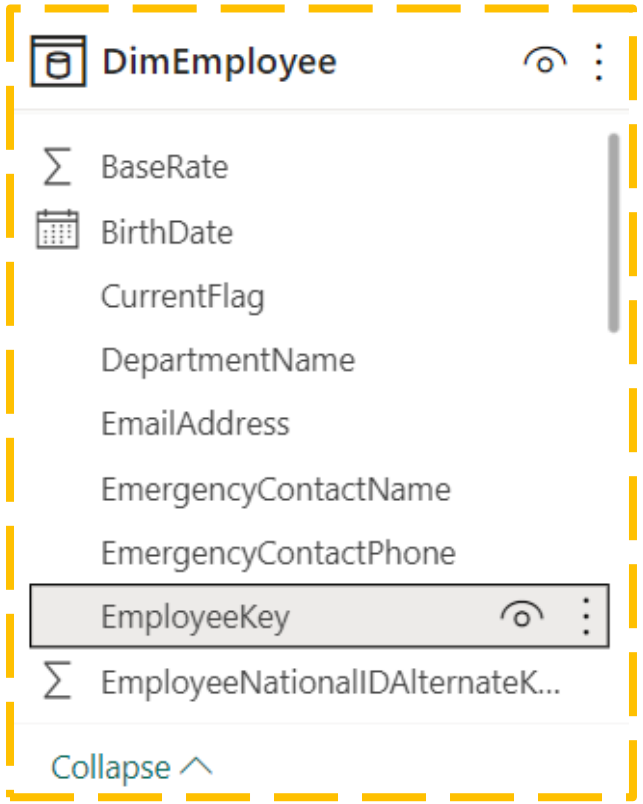
Introduction to data modeling

Data table types

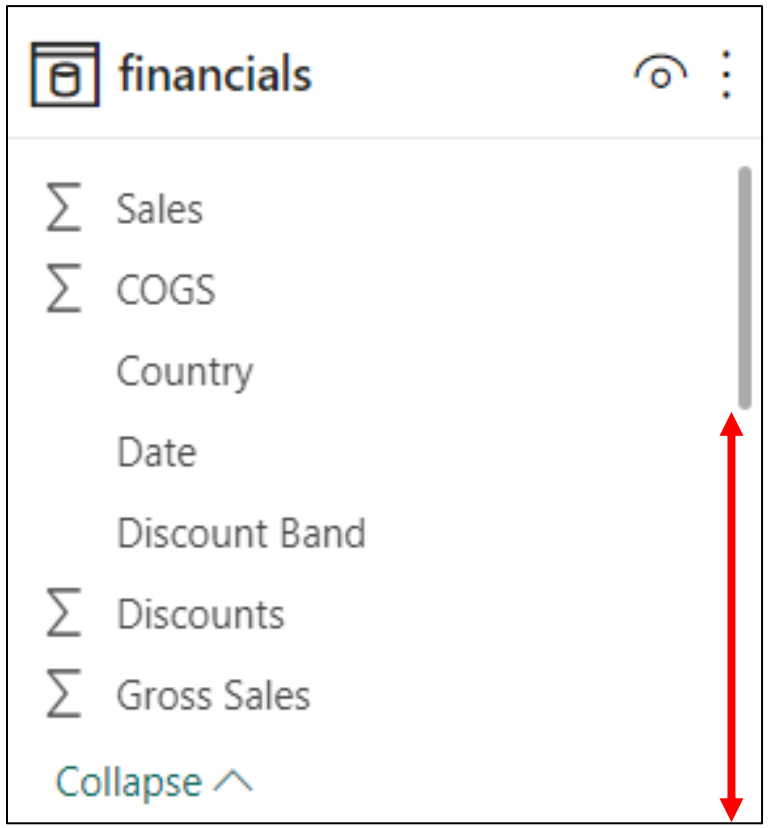
Fact tables are activities or events.

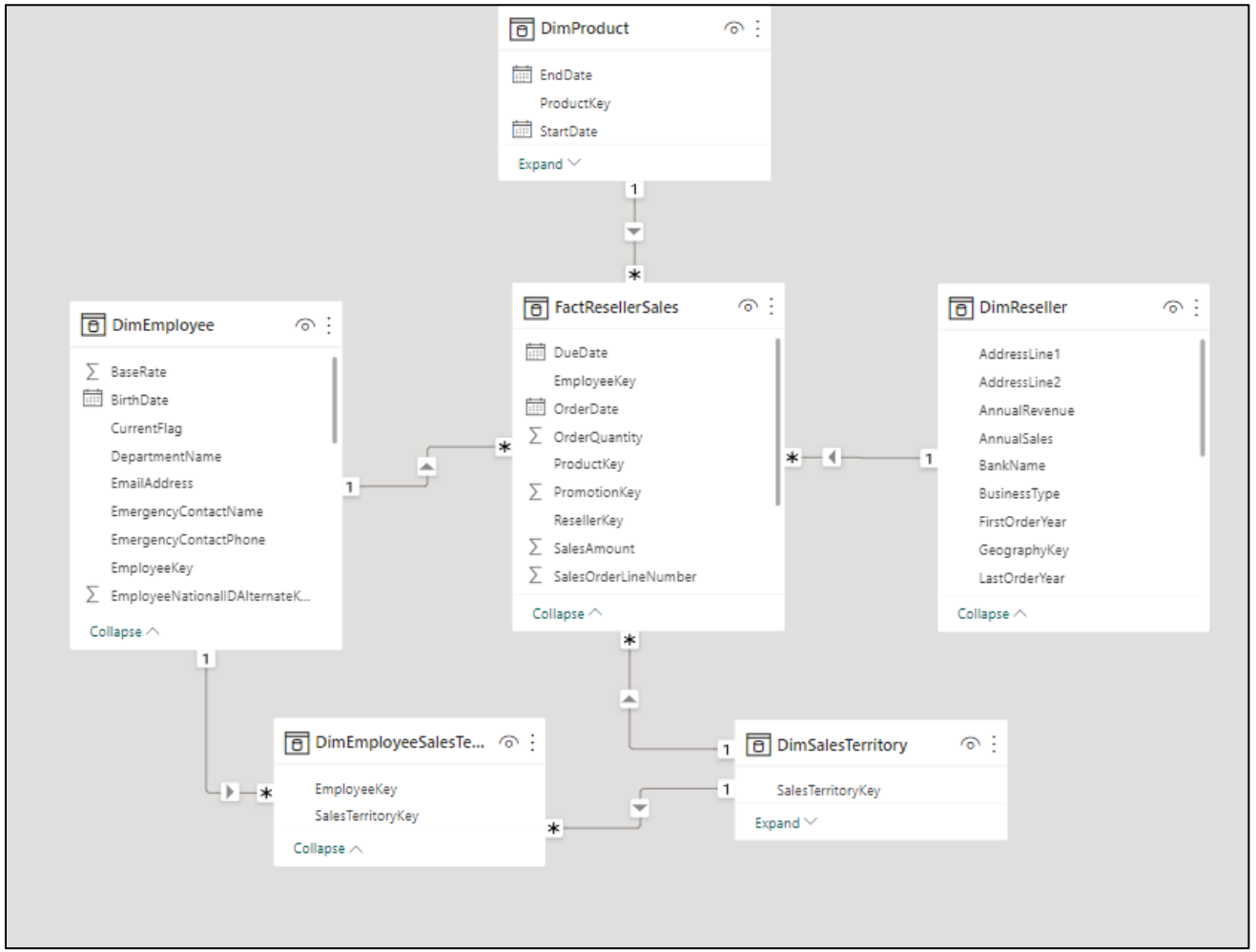


Dimension tables provide the details.



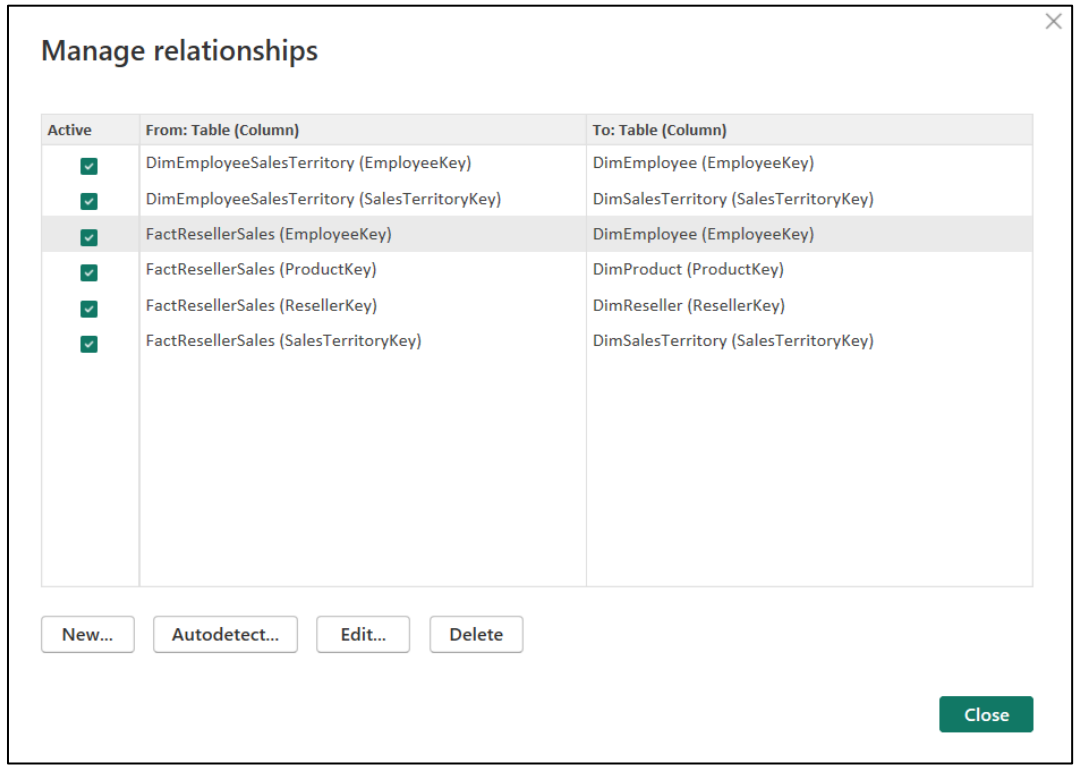
Understand star schemas

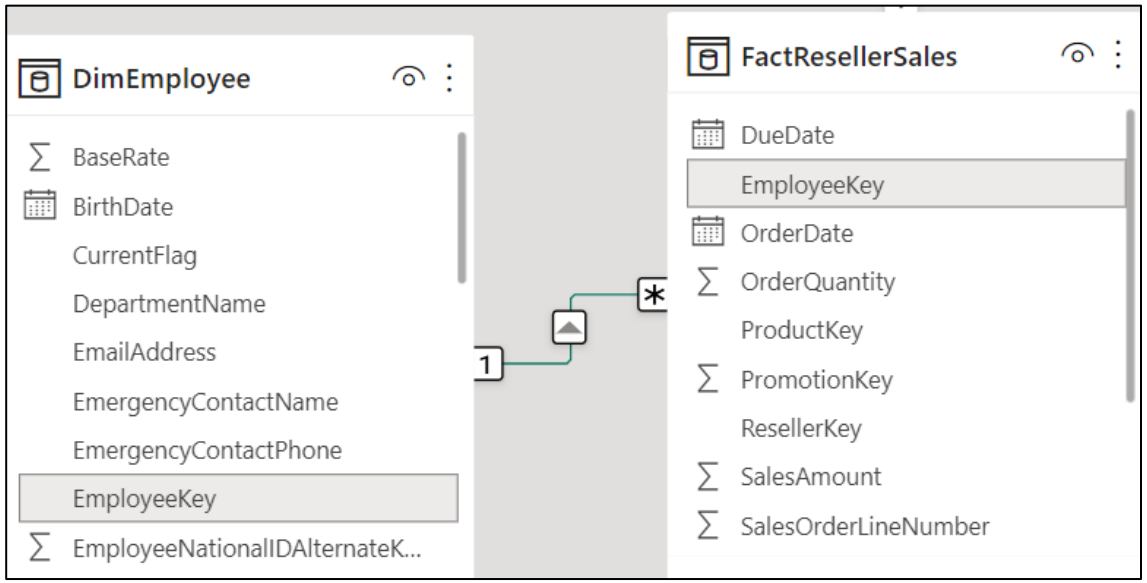




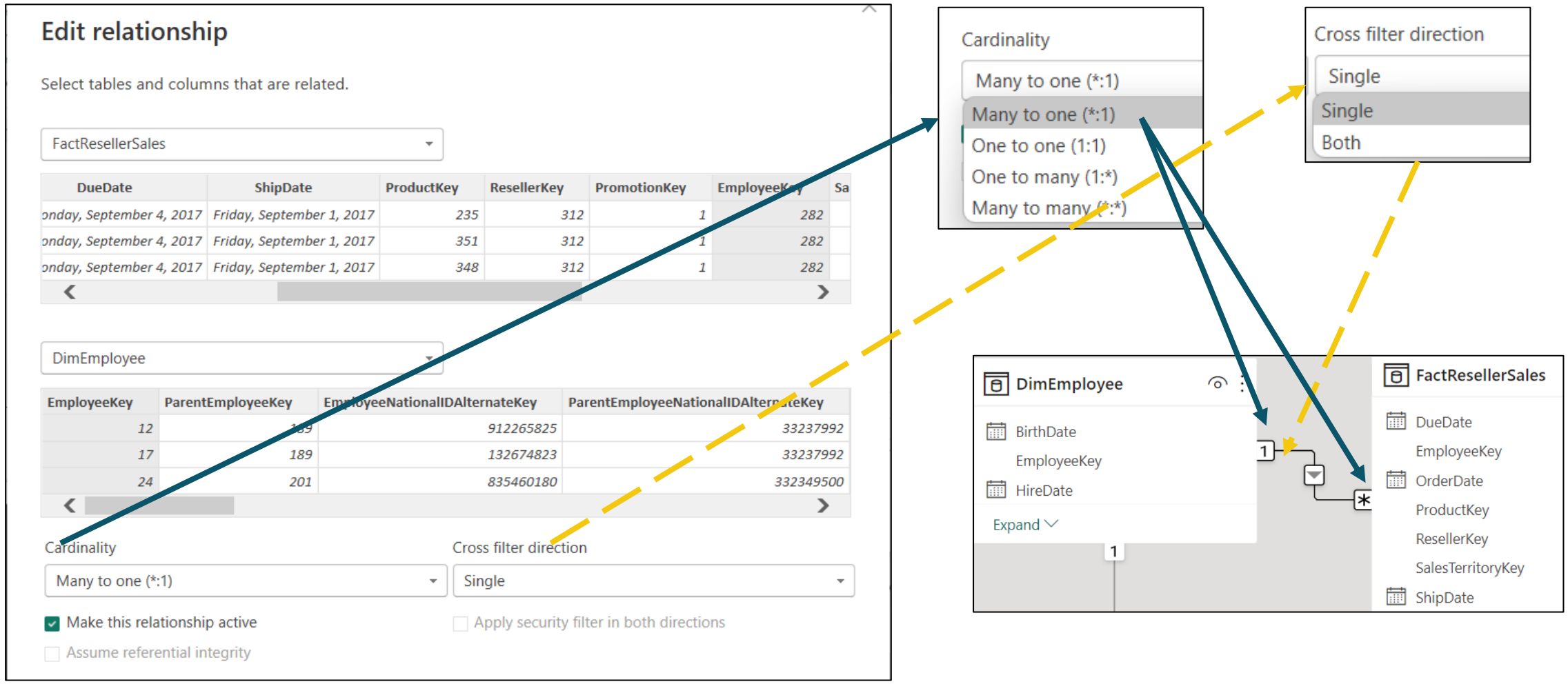
Manage relationships

Create relationships

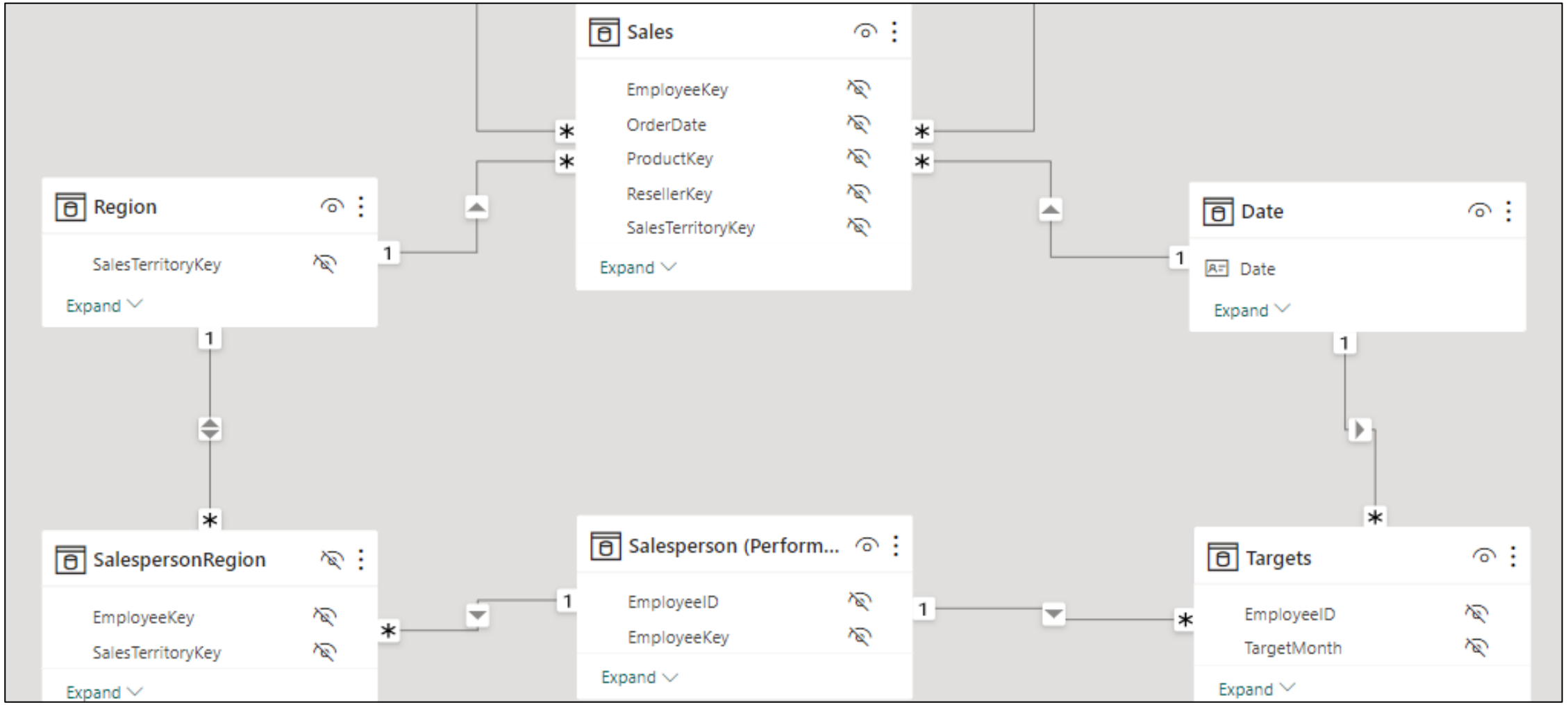




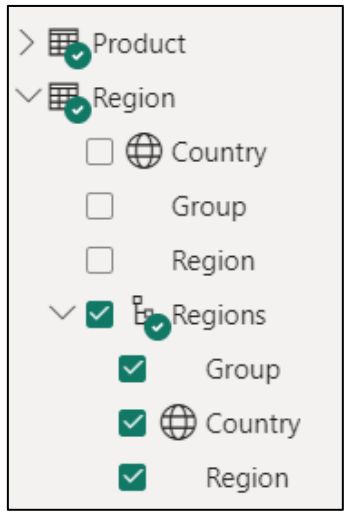
Edit relationships

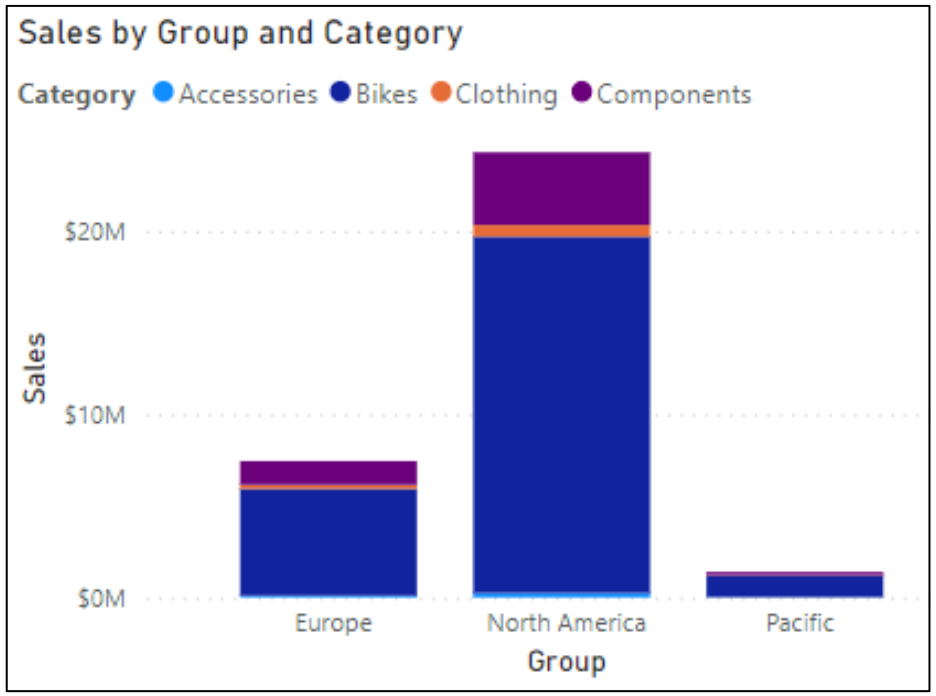


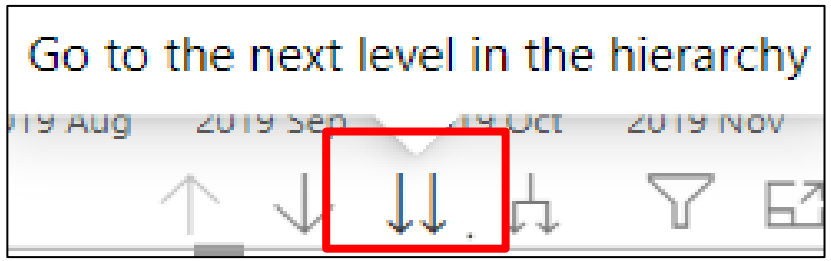
Implications of circular relationships

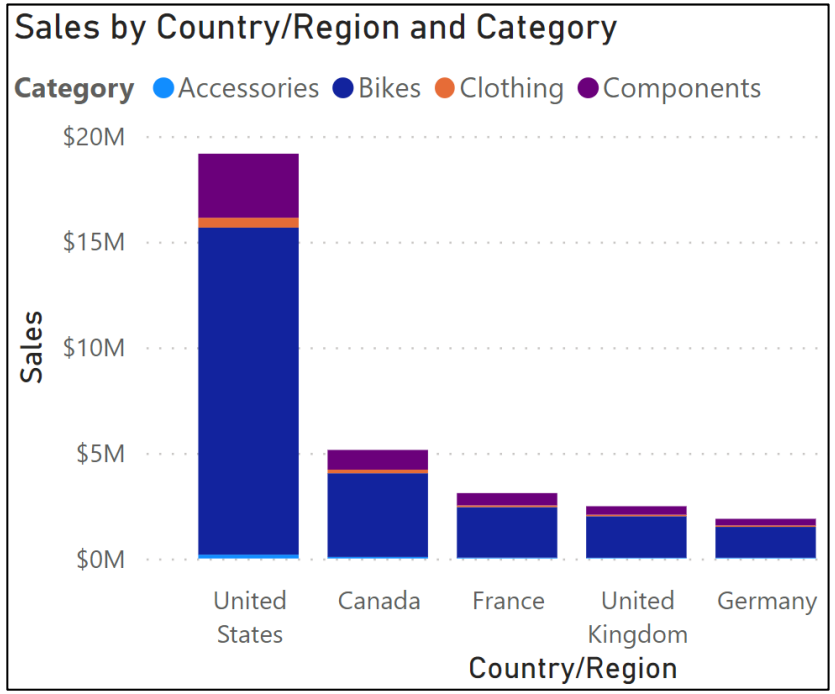


How to use hierarchies for data fields









Lab: Design a data model in Power BI Desktop

Configure data model - GitHub exercise

This lab should take approximately 45 minutes

In this lab, you'll commence developing the data model. It will involve creating relationships between tables, and then configuring table and column properties to improve the friendliness and usability of the data model.

- Create model relationships
- Configure table and column properties
- Create hierarchies and quick measures

Knowledge check: data models

What is the difference between a fact table and a dimension table?

- ☑ Fact tables contain observational data while dimension tables contain information about specific entities within the data.
- ☐ Fact tables contain information about specific entities while dimension tables contain information about observational data.
- ☐ There is no difference.

How do you describe a star schema?

- ☐ Fact and dimension tables with no relationships between them.
- ☐ Multiple fact tables with relationships to one dimension table.
- ☑ Multiple dimension tables with relationships to one fact table.

Recap

In this section, we covered:

Fact and dimension tables.

Designing a star schema layout.

How relationships connect tables.

Managing relationships.

Creating hierarchies.

Microsoft

Create calculations with DAX in Power BI

aka.ms/PL300-4

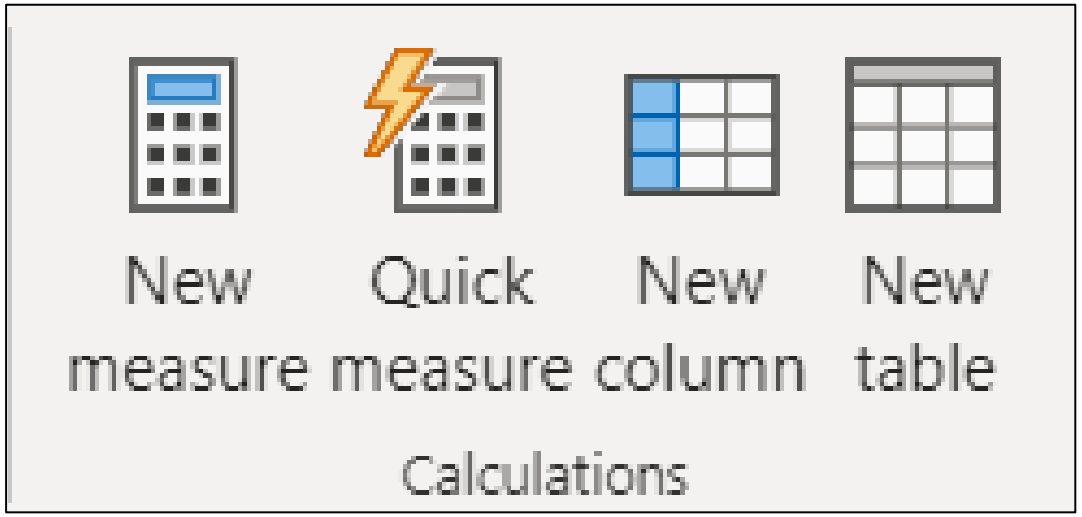
Agenda

- Introduction to DAX
- Advanced DAX concepts
- Optimize DAX performance

Introduction to DAX

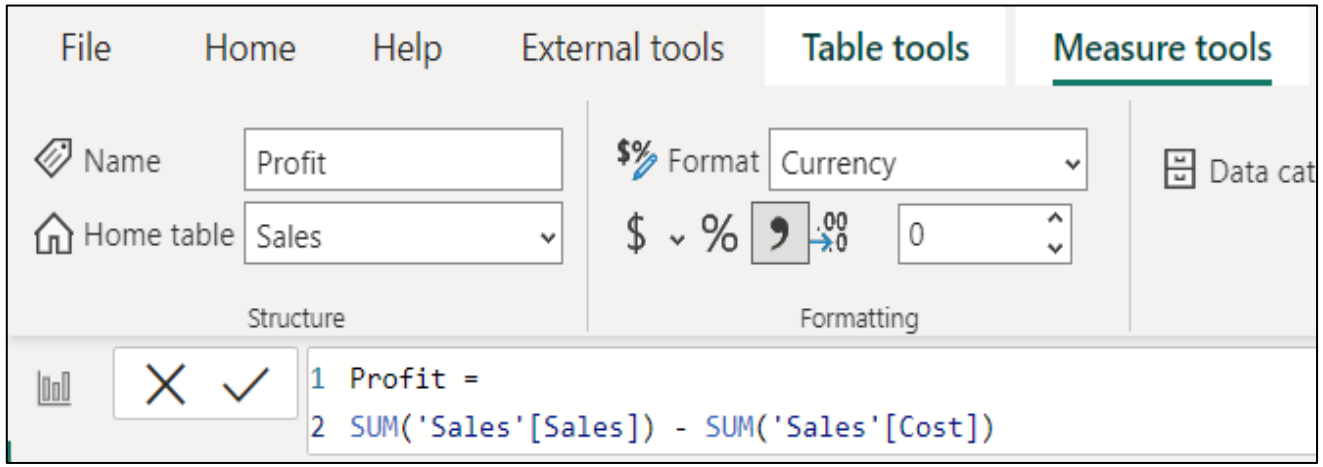
What is DAX?

- **D**ata **A**nalysis E**x**pressions
- Library of functions and operators
- Build formulas and expressions
- Create calculated measures, columns, and tables

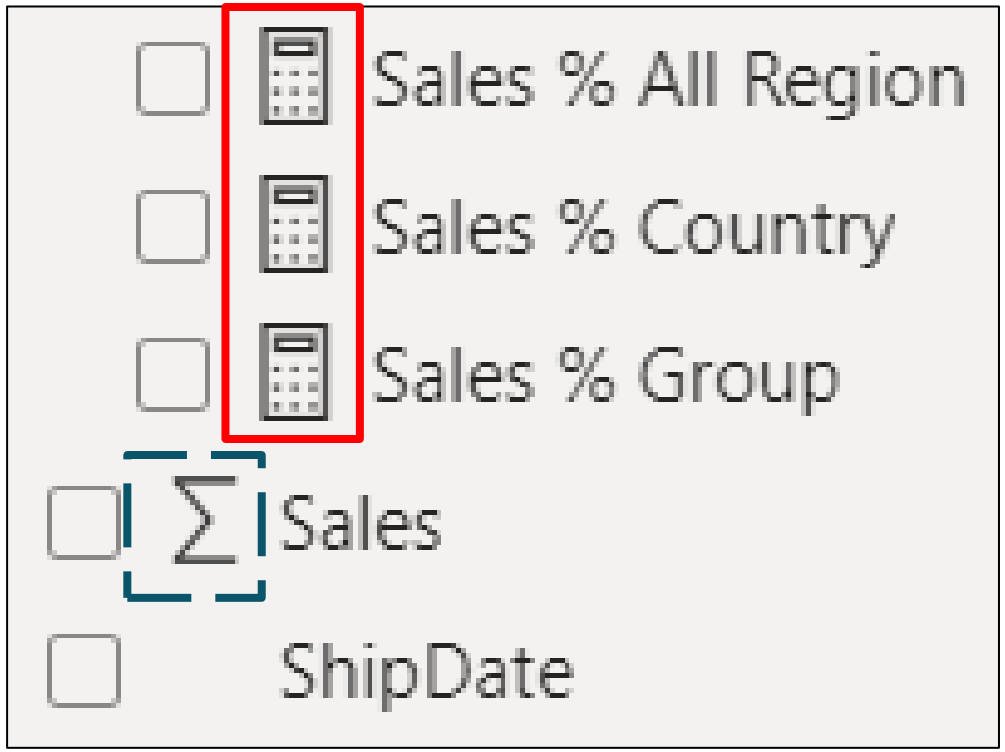


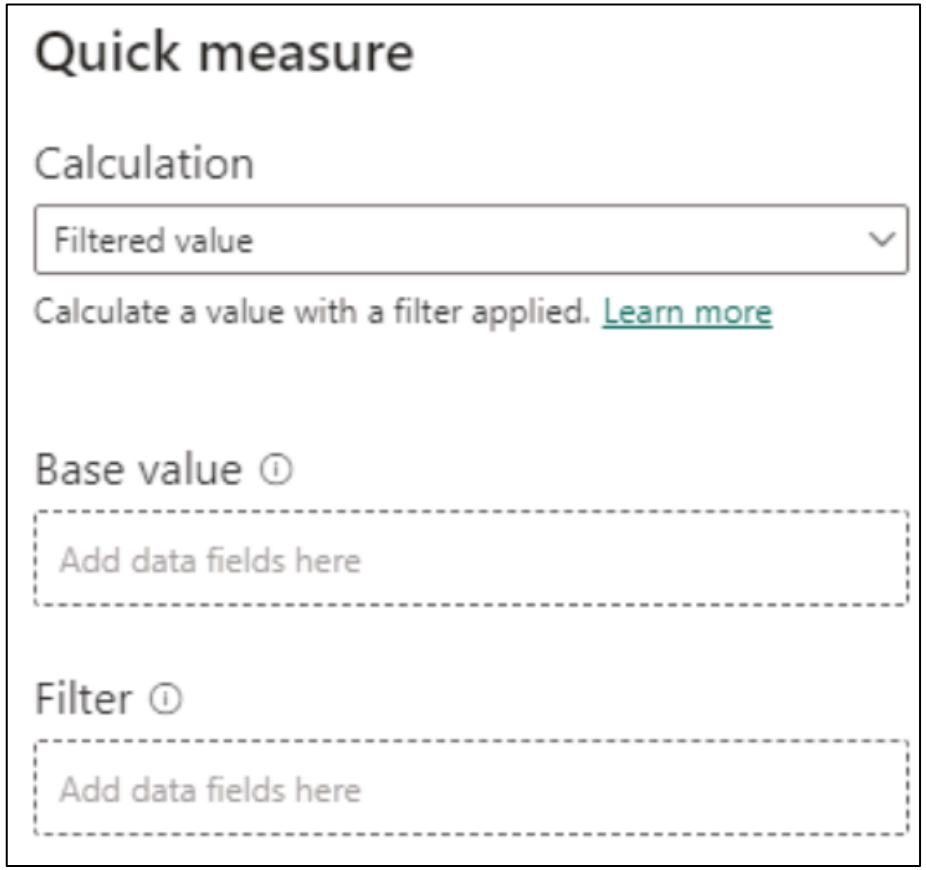
Create calculated measures

- Defined with DAX definitions
- Computed on the fly.
- Not stored in data model.
- Responsive to interactions.
- Indicated by calculator icon.



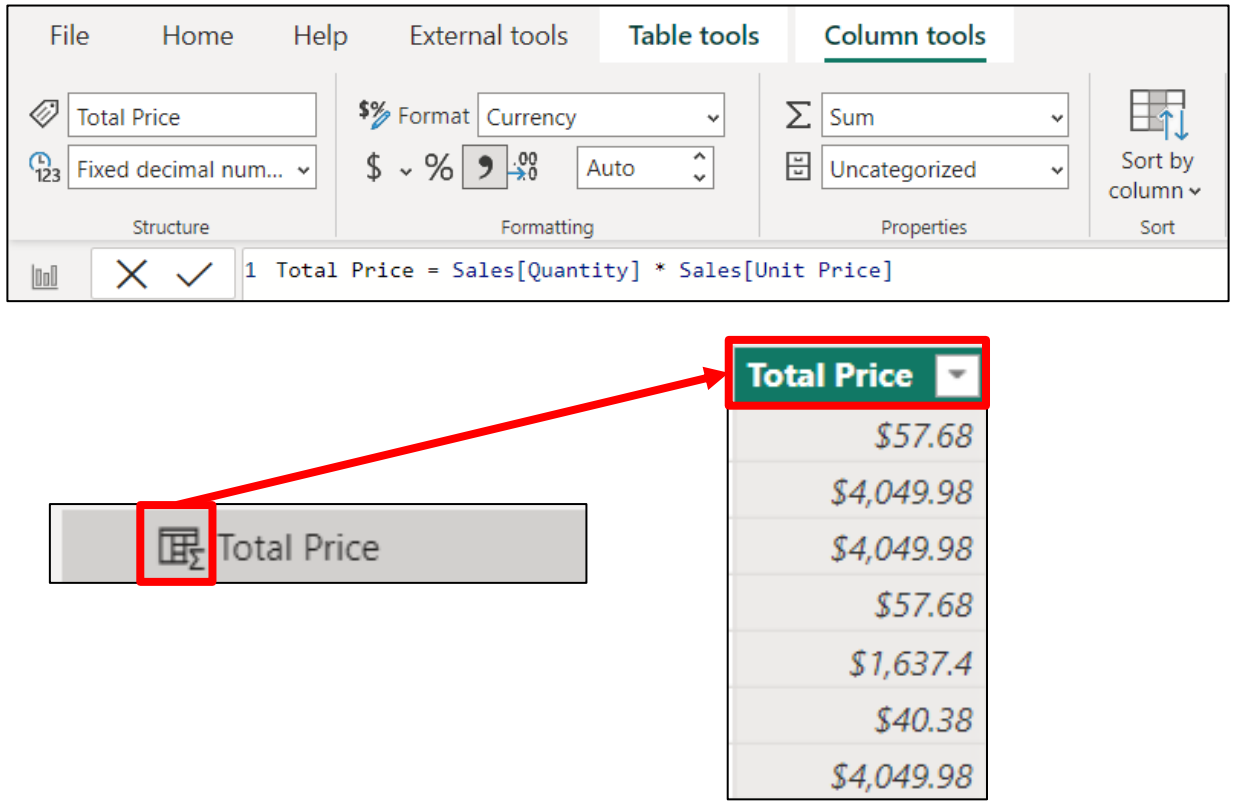
Implicit vs. Explicit vs. Quick measures





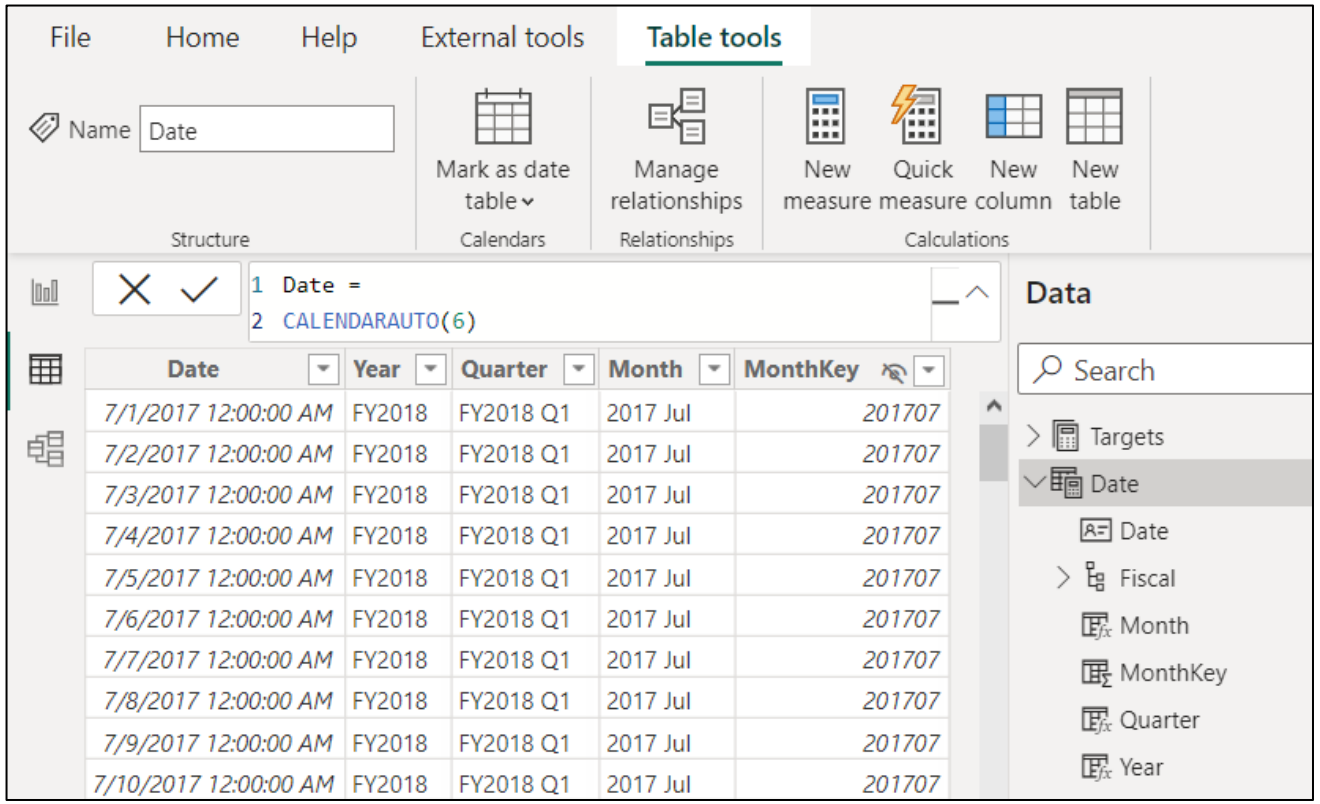
Create calculated columns

- Defined using DAX expressions.
- Computed & stored in data model.
- Useful “helper/connector columns.”
- Recalculated during data refresh.
- Table and Sigma icon.



Create calculated tables

- Defined using DAX expressions.
- Computed & stored in data model.
- Useful for aggregating data or creating custom tables.
- Table and calculator icon.



Columns vs. measures

Calculated columns:

- Create values for each row in table.
- Store values in the .pbix file.
- Increases data model size.
- Row-by-row calculation can impact performance.
- Must be referenced with measures for reuse.

Measures:

- Calculate on demand.
- Calculated based on filters.
- Doesn’t affect data model size.
- DAX expressions may still be suboptimal.
- Can reference other measures directly for reuse.

Lab: Create DAX calculations in Power BI Desktop (45 min)

Create DAX calculations | GitHub exercise

In this lab you'll create calculated tables, calculated columns, and simple measures using Data Analysis Expressions (DAX).

In this lab you learn how to:

- Create calculated tables
- Create calculated columns
- Create measures

Knowledge check: DAX calculations

Which statement about measures is correct?

- ☐ Measures store values in the data model.
- ☐ Measures can reference columns directly.
- ☑ Measures can reference other measures directly.

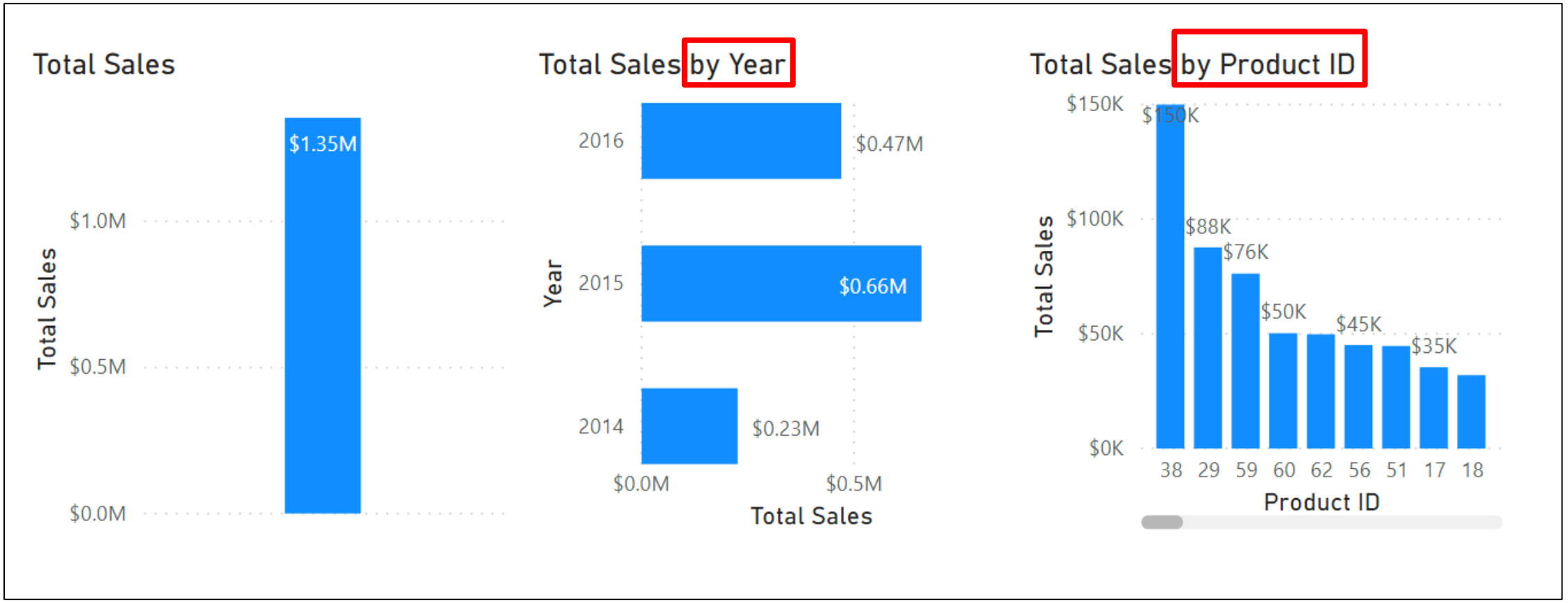
Which of the following statements describing similarity of measures and calculated columns in an Import model is true?

- ☑ They can achieve summarization of model data.
- ☐ They store values in the data model.
- ☐ They're evaluated during data refresh.

Advanced DAX concepts

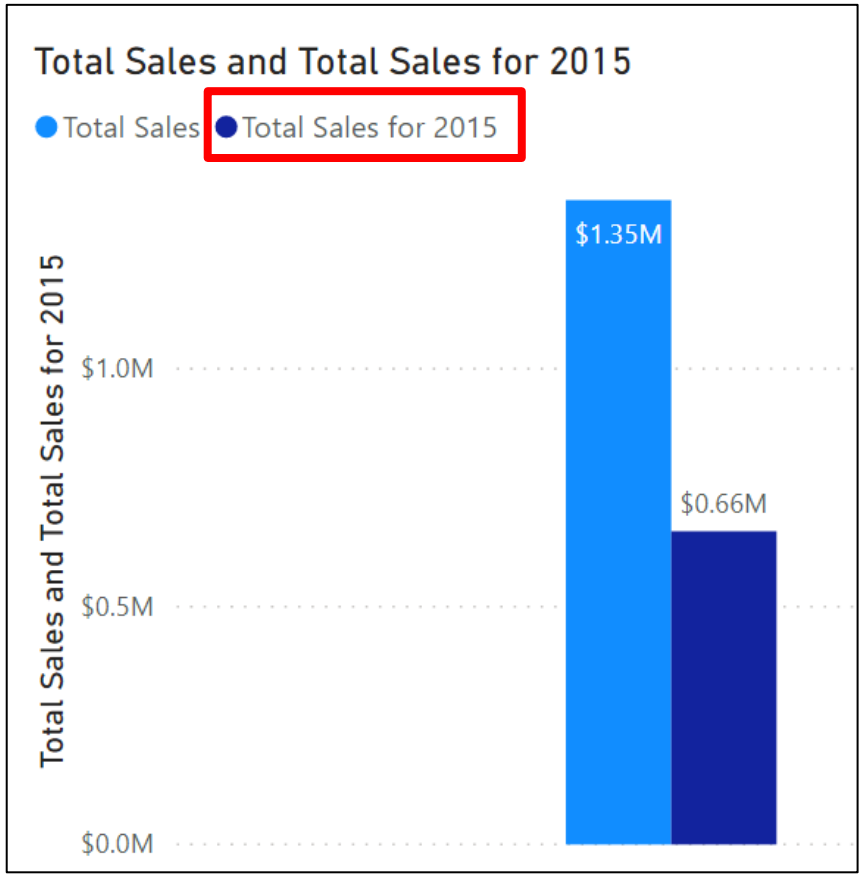
Understanding filter context

Measures are contextually different, or “dynamic,” depending on filters.



CALCULATE() function

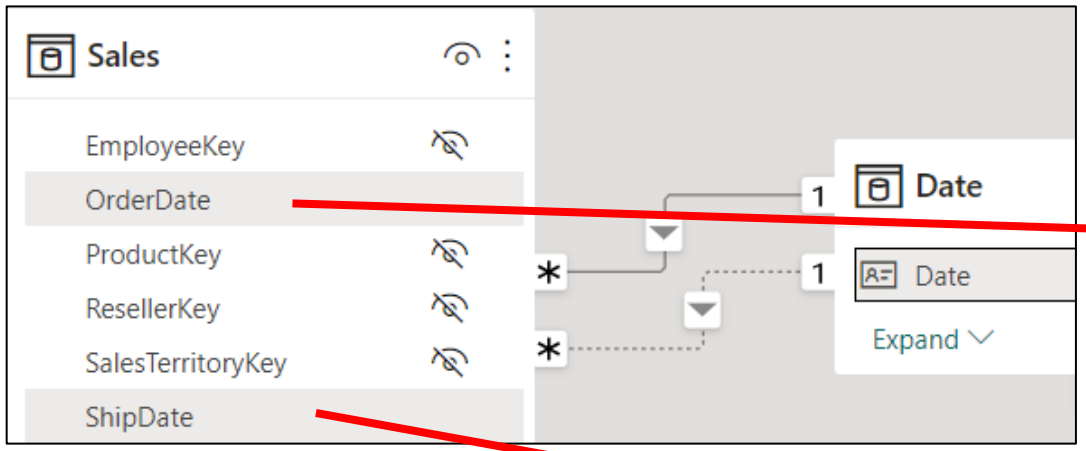
Adjusts how measures interpret data filters, enabling context control.

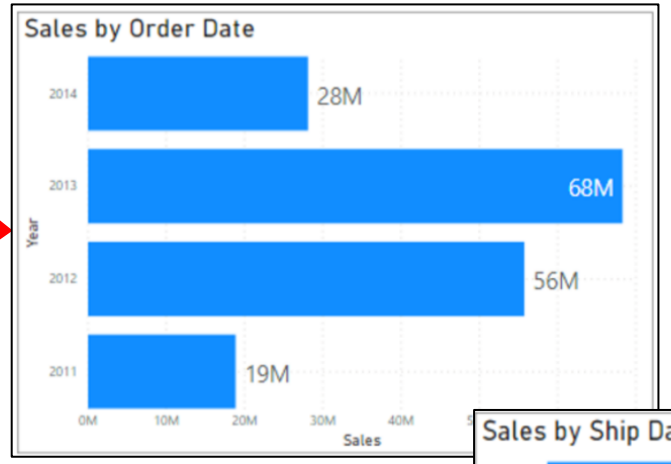


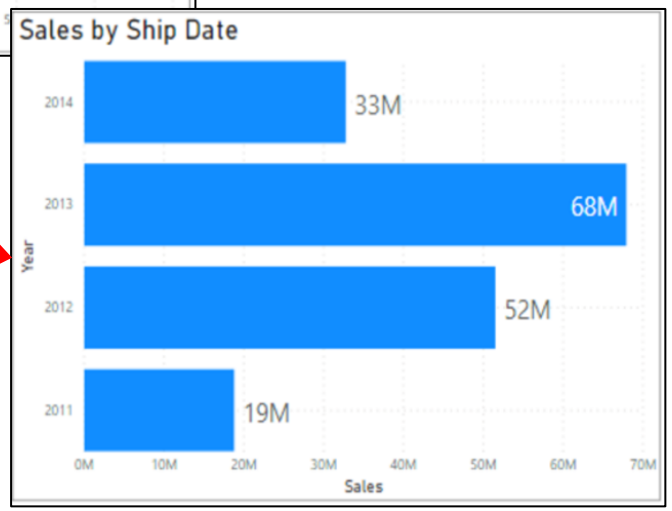

```
Total Sales for 2015 =
CALCULATE (
    SUM ( 'Sales OrderDetails'[Total Price]
),
    YEAR ( 'Sales OrderDetails’[OrderDate] )
= 2015
)
```

Use inactive relationships with DAX

Enable additional table filtering without impacting the active relationship.



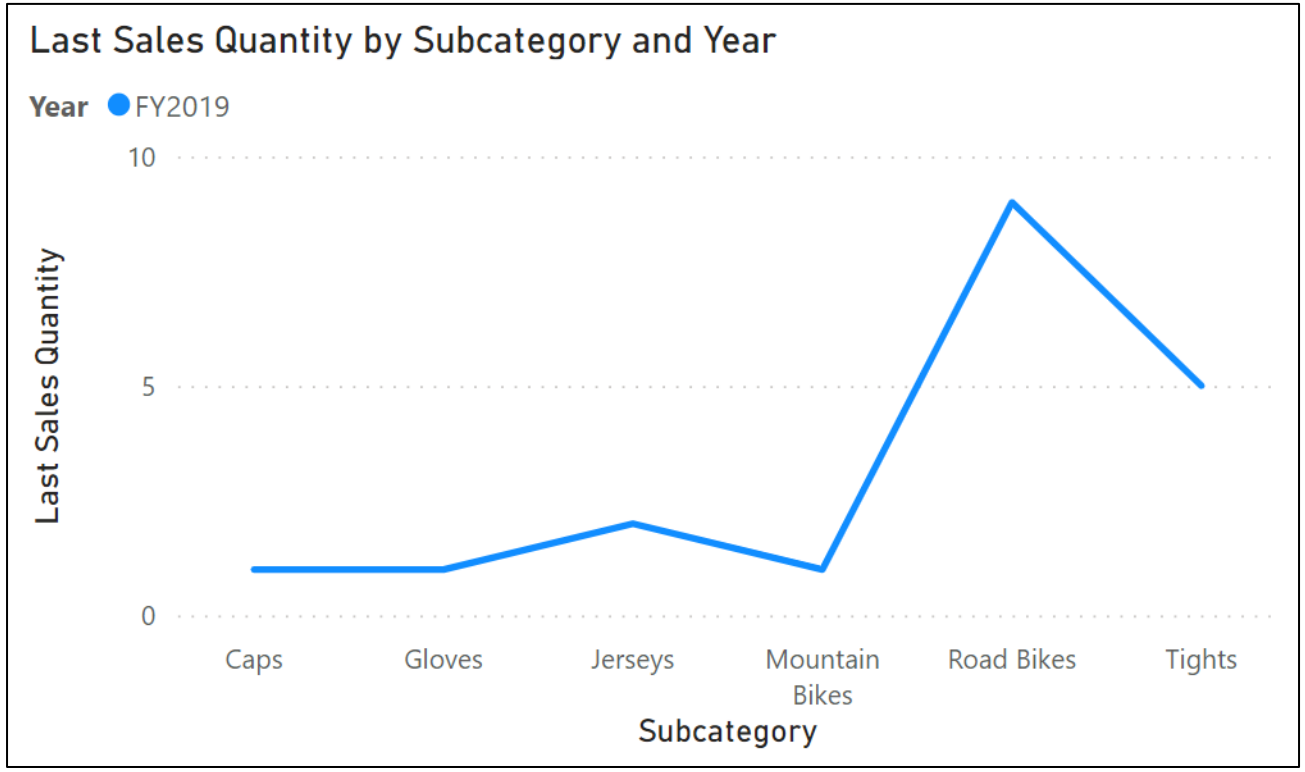





```
Sales by Ship Date =
CALCULATE (
    Sales[TotalPrice],
    USERELATIONSHIP (
       'Calendar'[Date], Sales[ShipDate])
)
```

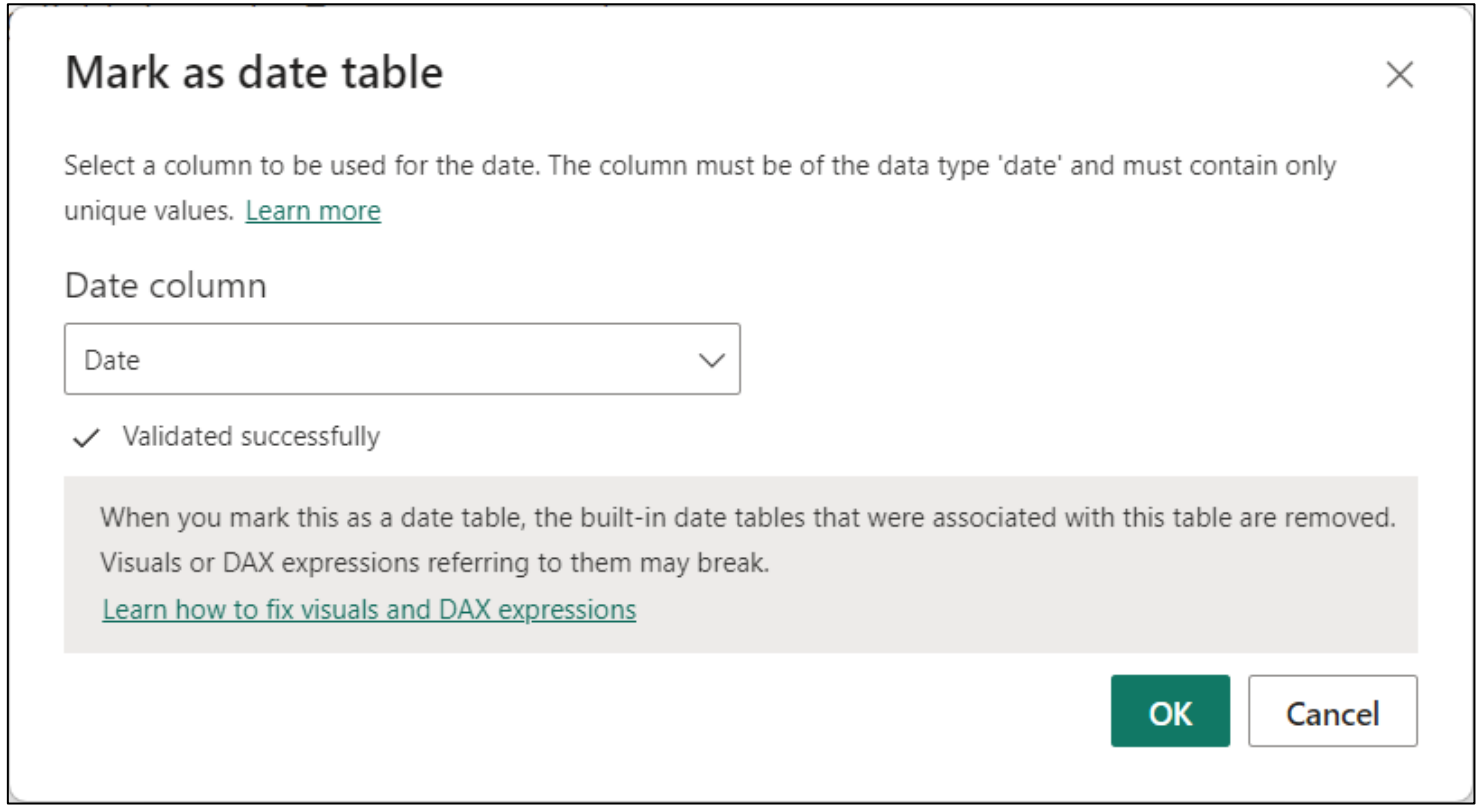
Semi-additive measures

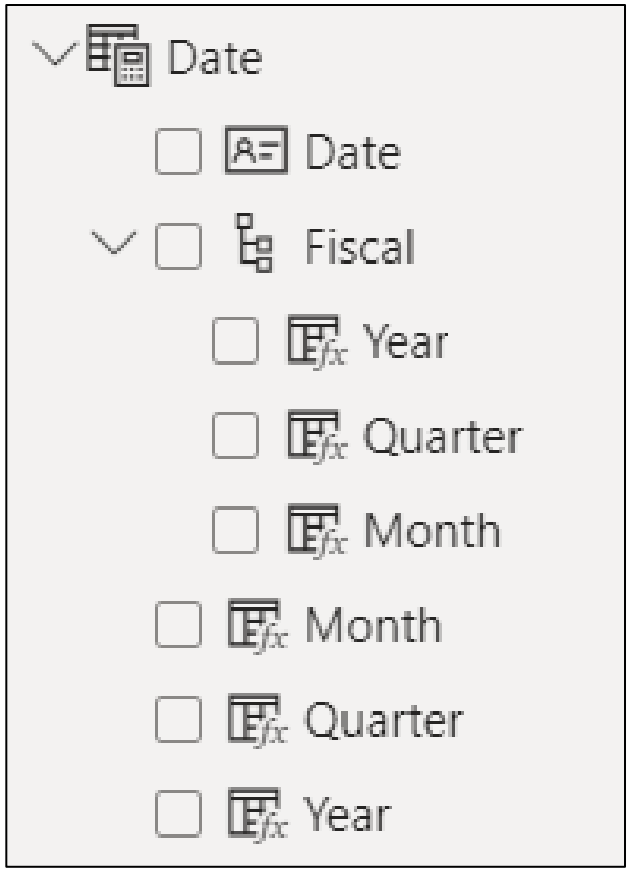
- **Context-dependent**: Results vary based on dimensions.
- **Time Intelligence**: Useful for averages, ratios, balances.
- **Caution with aggregation**: Beware of misleading results when aggregating.




```
Last Sales Quantity =
CALCULATE(
    SUM(Sales[Quantity]),
       LASTDATE('Date'[Date])
)
```

Create a common date table





Time Intelligence functions

Month	2014	2015	2016
January		$66,692.8	$100,854.72
February		$107,900	$205,416.67
March		$147,879.9	$315,242.12
April		$203,579.29	$449,872.68
May		$260,402.99	$469,771.34
June		$299,490.99	$469,771.34
July	$30,192.1	$354,955.92	$469,771.34
August	$56,801.5	$404,937.61	$469,771.34
September	$84,437.5	$464,670.63	$469,771.34
October	$125,641.1	$534,999.13	$469,771.34
November	$175,345.1	$580,912.49	$469,771.34
December	$226,298.5	$658,388.75	$469,771.34
Total	**$226,298.5**	**$658,388.75**	**$469,771.34**

```
Total Sales Previous Month =

CALCULATE ( SUM ( Sales[Total Price] ),
PREVIOUSMONTH ( 'Date'[Date] ) )
```

Year	Month	Total Sales	Total Sales Previous Month
2015	March	$39,979.9	$41,207.2
2015	April	$55,699.39	$39,979.9
2015	May	$56,823.7	$55,699.39
2015	June	$39,088	$56,823.7
2015	July	$55,464.93	$39,088
2015	August	$49,981.69	$55,464.93
2015	September	$59,733.02	$49,981.69
2015	October	$70,328.5	$59,733.02
2015	November	$45,913.36	$70,328.5
2015	December	$77,476.26	$45,913.36

Lab: Advanced DAX calculations in Power BI Desktop (45 min)

Advanced DAX | GitHub Exercise

In this lab, you'll create measures with DAX expressions involving filter context manipulation.

In this lab you learn how to:

- Use the CALCULATE() function to manipulate filter context
- Use Time Intelligence functions

Knowledge check: advanced DAX

What DAX function lets you use inactive relationships?

- ☑ USERELATIONSHIP
- ☐ DEACTIVATEFILTERS
- ☐ OVERRIDECONNECTION

Which type of functions specifically use dates for calculations?

- ☐ Date and Time
- ☑ Time Intelligence
- ☐ Filter

What kind of measure sums for one dimension and uses another aggregation for a different dimension?

- ☐ Summarized
- ☐ Aggregated
- ☑ Semi-additive

Optimize DAX performance

Use variables to improve performance and readability

Without variable:

```
Sales YoY Growth =
   DIVIDE (
       ( [Sales] - CALCULATE ( [Sales], PARALLELPERIOD ( 'Date'[Date], -12, MONTH ) ) ),
       CALCULATE ( [Sales], PARALLELPERIOD ( 'Date'[Date], -12, MONTH ) ) )
```

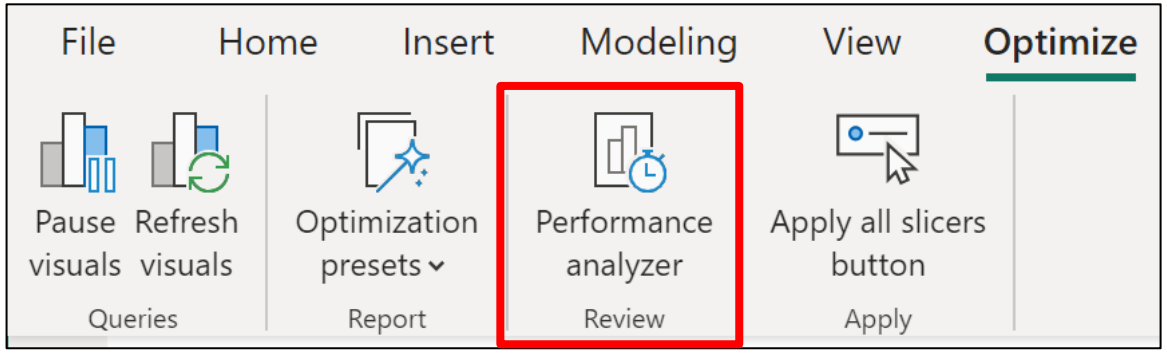
With variable:

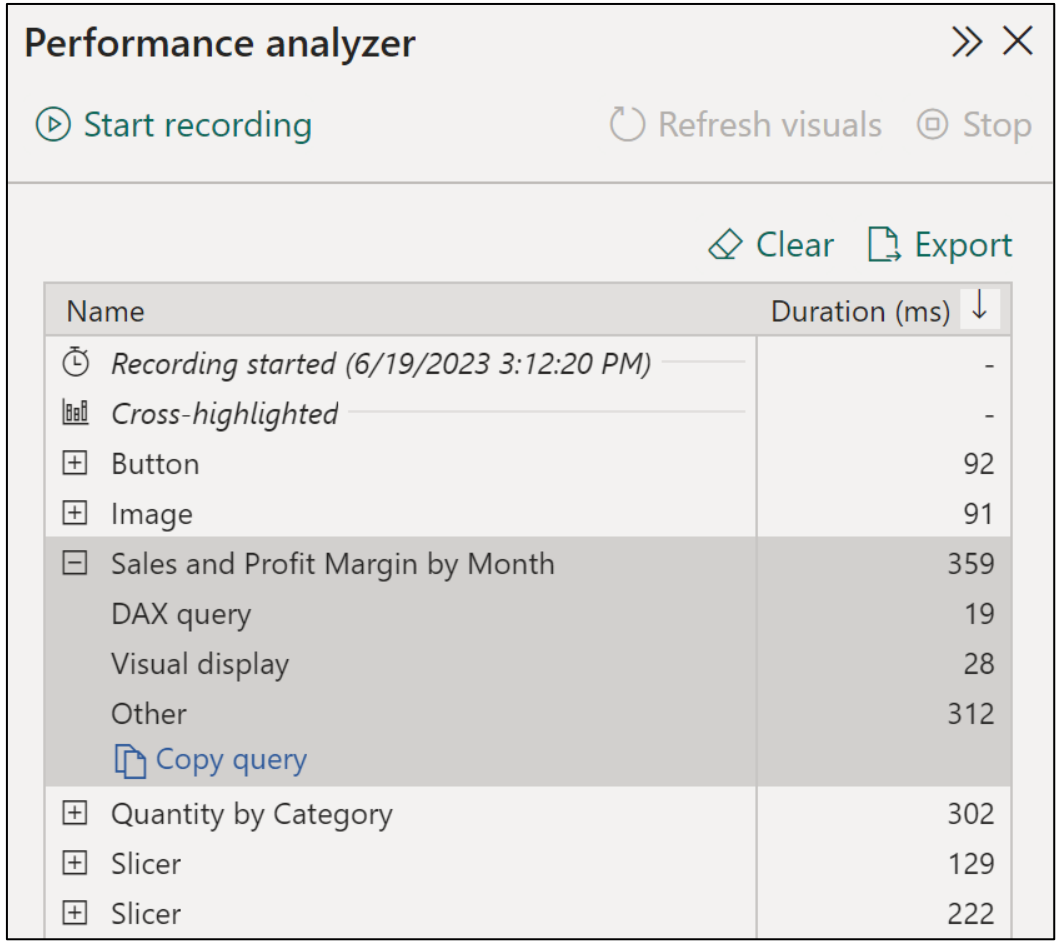
```
Sales YoY Growth =
    VAR SalesPriorYear = CALCULATE ( [Sales], PARALLELPERIOD ( 'Date'[Date], -12, MONTH ) )
    VAR SalesVariance =  DIVIDE ( ( [Sales] - SalesPriorYear ), SalesPriorYear )
  RETURN
  SalesVariance
```

Tune report performance

Performance analyzer in Power BI Desktop

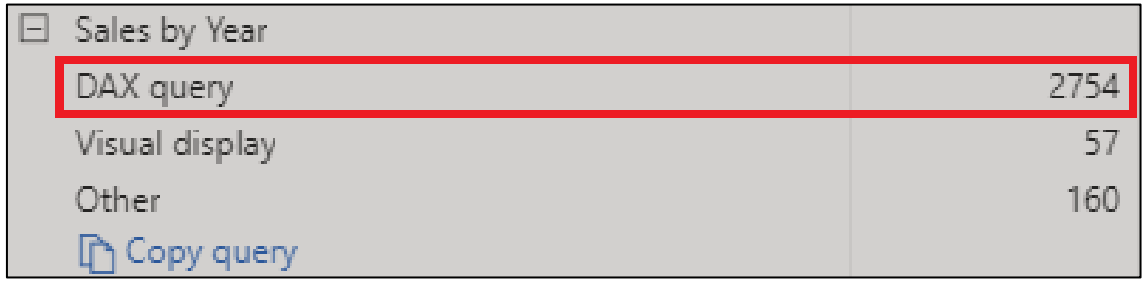
- Record report interactions.
- Review query and rendering times.
- Identify bottlenecks, optimize queries.



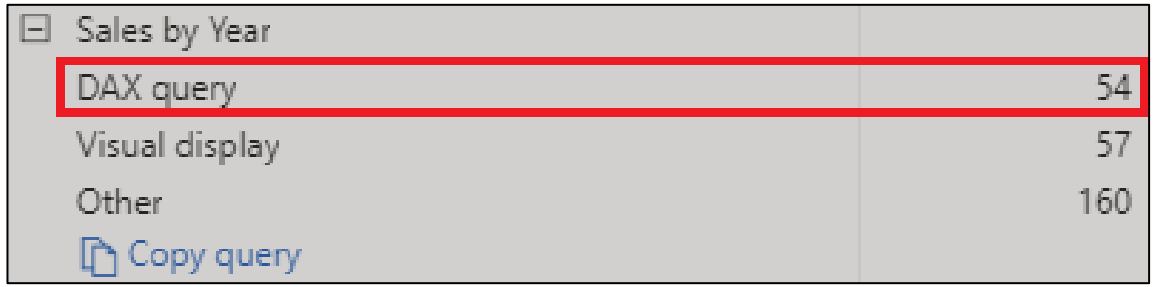


Analyze query plans for optimization opportunities

```
Count Customers =
CALCULATE (
    DISTINCTCOUNT ( Order[ProductID] ),
    FILTER ( Order, Order[OrderQty] >= 5 )
)
```




```
Count Customers =
CALCULATE (
    DISTINCTCOUNT ( Order[ProductID] ),
    KEEPFILTERS (Order[OrderQty] >= 5 )
)
```



Knowledge check: DAX optimization

What benefit do you get from analyzing the metadata?

- ☑ The benefit of analyzing the metadata is that you can clearly identify data inconsistences with your semantic model.
- ☐ The benefit of analyzing the metadata is to get familiar with your data.
- ☐ The benefit of analyzing the metadata is to know the number of rows, columns and tables being loaded into your model.

Which of the following statements about relationships in Power BI Desktop is true?

- ☑ Relationships can only be created between columns that contain the same data type.
- ☐ Relationships can only be created between tables that contain the same number of rows.
- ☐ Relationships can be created between tables that contain different types of data.

Recap

In this section, we covered:

What DAX is and how to use it.

Creating calculated measures, columns, and tables with DAX.

Benefits of measures vs. calculated columns.

How filter context and CALCULATE() function affect measures.

Inactive relationships and semi-additive measures.

Creating a common date table and Time Intelligence functions.

Using variables and Performance analyzer for performance.

Microsoft

Thanks

Resources

- Design a data model in Power BI
- Write DAX formulas for Power BI Desktop models
- Add measures to Power BI Desktop models
- Use DAX time intelligence functions in Power BI Desktop models