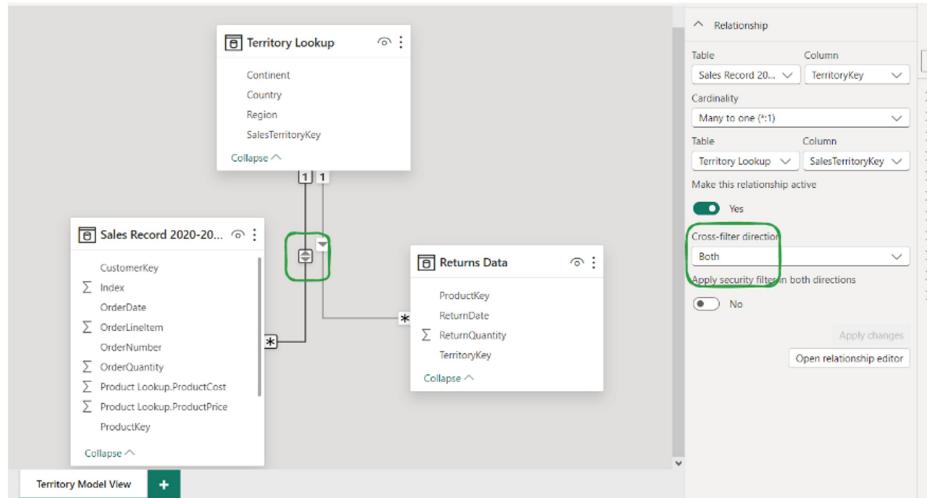


Power BI - Data Modelling - p3

Power BI - Data Modelling - p3



Territory Lookup

SalesTerritoryKey	Sum of OrderQuantity	Sum of ReturnQuantity
1	12513	270
2	40	
3	30	
4	17191	362
5	49	1
6	10894	238
7	7862	186
8	7950	163
9	17951	404
10	9694	204
Total	84174	1828

Sales Record - Cross Filter - Correct Answer

TerritoryKey	Sum of OrderQuantity	Sum of ReturnQuantity
1	12513	270
2	40	
3	30	
4	17191	362
5	49	1
6	10894	238
7	7862	186
8	7950	163
9	17951	404
10	9694	204
Total	84174	1828

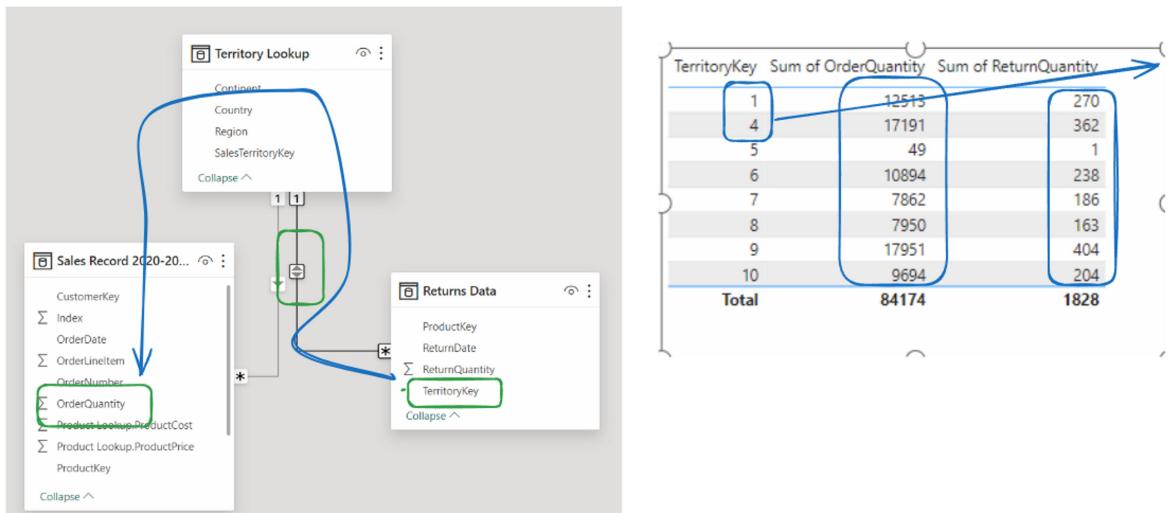
TerritoryKey	Sum of OrderQuantity	Sum of ReturnQuantity
1	84174	270
4	84174	362
5	84174	1
6	84174	238
7	84174	186
8	84174	163
9	84174	404
10	84174	204
Total	84174	1828

Error

There are ambiguous paths between 'Territory Lookup' and 'Product Lookup': 'Territory Lookup'->'Sales Record 2020-2022'->'Product Lookup' and 'Territory Lookup'-'>'Returns Data'-'>'Product Lookup'

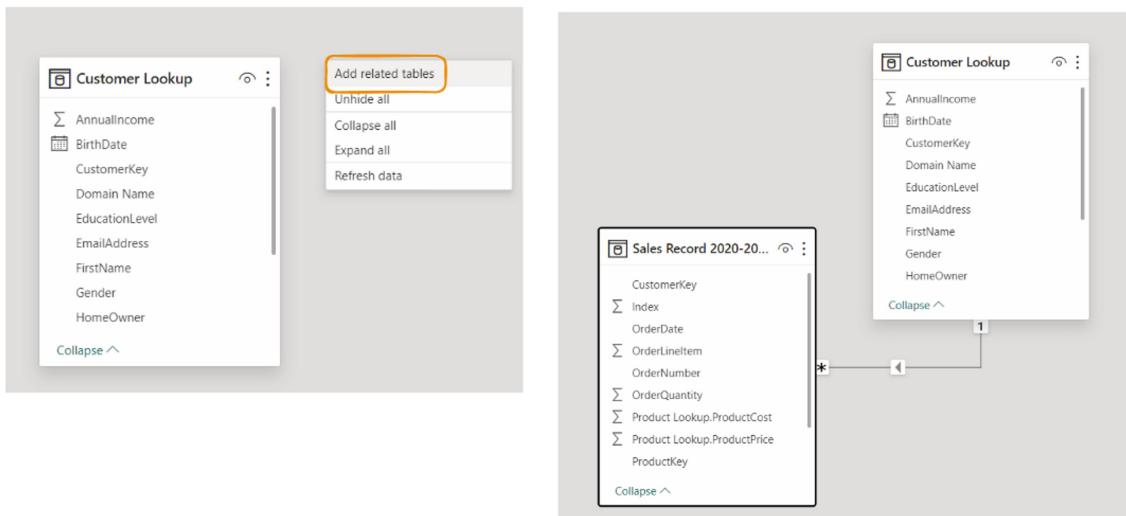
Close

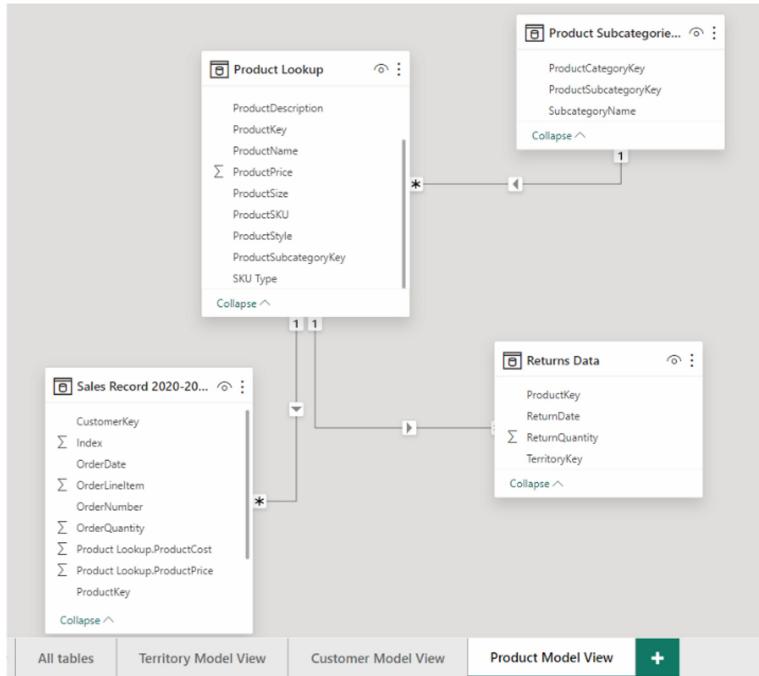
Return Table



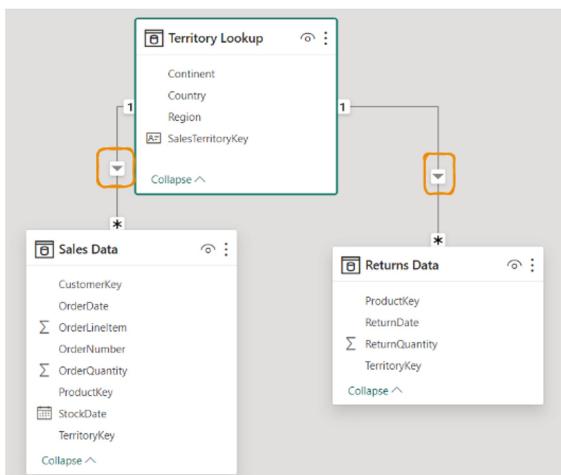
As we progress, it would be difficult to handle wrong insight coming from bi-direction/ Cross Filter.

Never ever pick foreign key to find an insight.





FILTER CONTEXT & FLOW



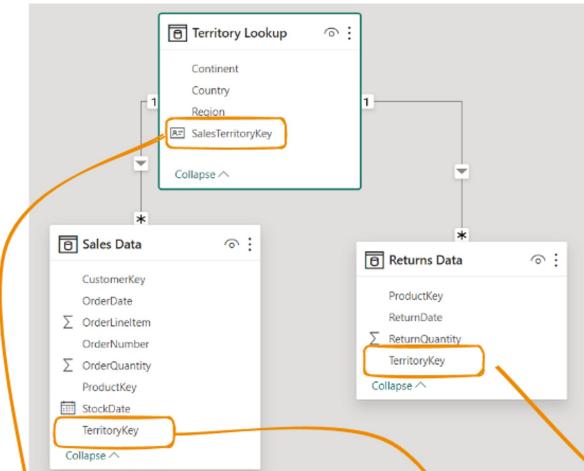
Here we have two data tables (Sales Data and Returns Data), connected to Territory Lookup

The arrows show the filter direction, and point from the one (1) side of the relationship to the many (*) side

- When you filter a table, that filter context is passed to any related "downstream" tables, following the arrow's direction

- Filter context CANNOT flow "upstream"

PRO TIP: Arrange lookup tables above fact tables in your model as a visual reminder that filters always flow downstream



In this model, the only way to filter both Sales and Returns data by Territory is to use the Territory Key from the lookup table, which is upstream and related to both fact tables.

- Filtering using Territory Key from the Sales table yields incorrect Returns values, since the filter context can't flow to any other table

- Filtering using Territory Key from the Returns table yields incorrect Sales values, and is limited to territories that exist in the returns table

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	270
2	40	
3	30	
4	17,191	362
5	49	1
6	10,694	238
7	7,862	186
8	7,950	163
9	17,951	404
10	9,694	204
Total	84,174	1,828

Filtering by Territory Lookup[Territory Key]

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	1,828
2	40	1,828
3	30	1,828
4	17,191	1,828
5	49	1,828
6	10,694	1,828
7	7,862	1,828
8	7,950	1,828
9	17,951	1,828
10	9,694	1,828
Total	84,174	1,828

Filtering by Sales Data[Territory Key]

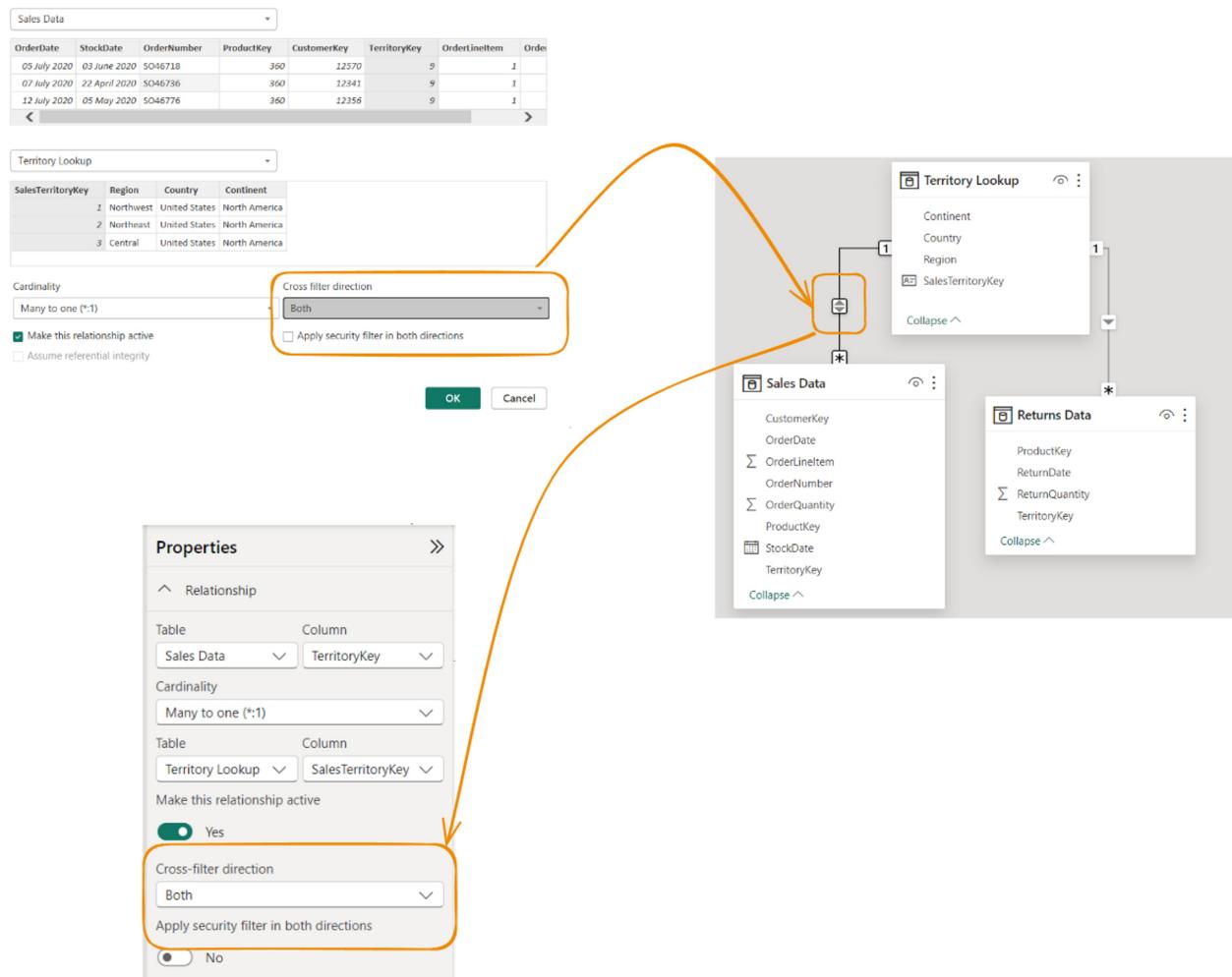
TerritoryKey	OrderQuantity	ReturnQuantity
1	84,174	270
4	84,174	362
5	84,174	1
6	84,174	238
7	84,174	186
8	84,174	163
9	84,174	404
10	84,174	204
Total	84,174	1,828

Filtering by Returns Data[Territory Key]

BI-DIRECTIONAL FILTERS

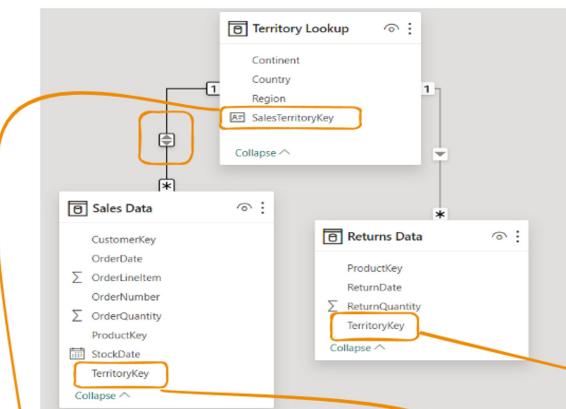
Edit relationship

Select tables and columns that are related.



Updating the cross-filter direction from Single to Both allows filter context to flow in either direction

- In this example, filters applied to the Sales table can pass up to the Territory Lookup table, then down to Returns



With two-way cross-filtering enabled between Sales and Territory, we now see correct values using Territory Key from either table

- Filter context can now pass up to the Territory Lookup table, then downstream to Returns

- However, we still see incorrect values when filtering using Territory Key from the Returns table, since the filter context is isolated to that single table

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	270
2	40	
3	30	
4	17,191	362
5	49	1
6	10,894	238
7	7,862	186
8	7,950	163
9	17,951	404
10	9,694	204
Total	84,174	1,828

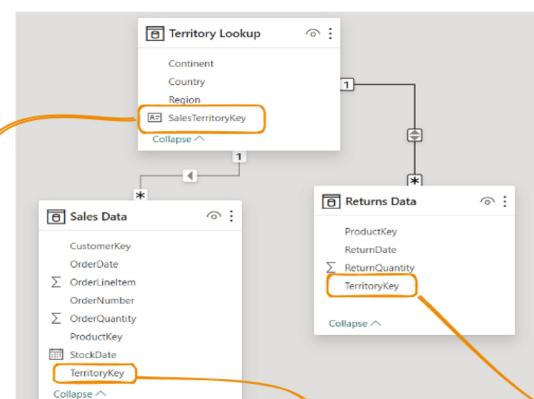
Filtering by Territory Lookup[Territory Key]

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	270
2	40	
3	30	
4	17,191	362
5	49	1
6	10,894	238
7	7,862	186
8	7,950	163
9	17,951	404
10	9,694	204
Total	84,174	1,828

Filtering by Sales Data[Territory Key]

TerritoryKey	OrderQuantity	ReturnQuantity
1	84,174	270
4	84,174	362
5	84,174	1
6	84,174	238
7	84,174	186
8	84,174	163
9	84,174	404
10	84,174	204
Total	84,174	1,828

Filtering by Returns Data[Territory Key]



In this case, we've enabled two-way cross-filtering between the Returns and Territory tables

- As expected, we now see incorrect values when filtering using Territory Key from the Sales table, since the filter context is isolated to that single table

- While the values appear to be correct when filtering using Territory Key from the Returns table, we're missing sales data from any territories that didn't appear in the returns table (specifically Territories 2 & 3)

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	270
2	40	
3	30	
4	17,191	362
5	49	1
6	10,894	238
7	7,862	186
8	7,950	163
9	17,951	404
10	9,694	204
Total	84,174	1,828

Filtering by Territory Lookup[Territory Key]

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	1,828
2	40	1,828
3	30	1,828
4	17,191	1,828
5	49	1,828
6	10,894	1,828
7	7,862	1,828
8	7,950	1,828
9	17,951	1,828
10	9,694	1,828
Total	84,174	1,828

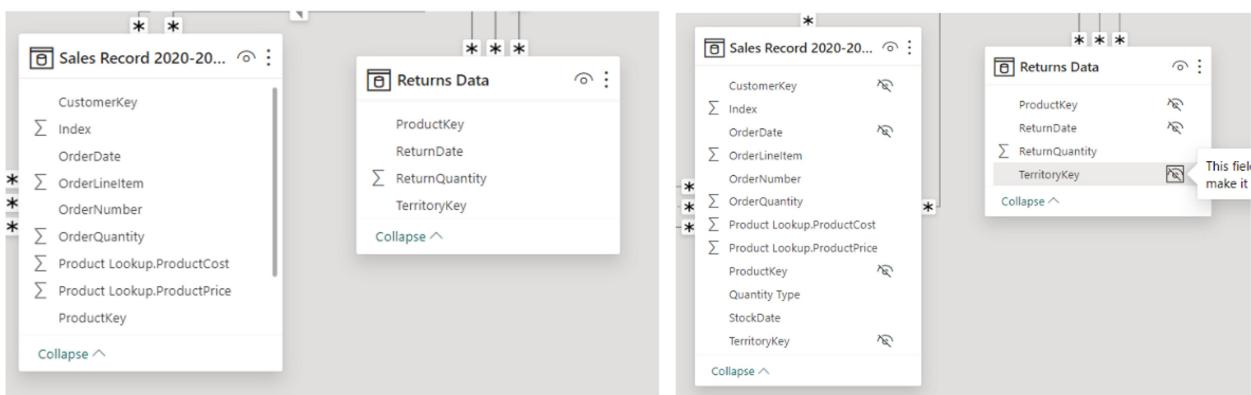
Filtering by Sales Data[Territory Key]

TerritoryKey	OrderQuantity	ReturnQuantity
1	12,513	270
4	17,191	362
5	49	1
6	10,894	238
7	7,862	186
8	7,950	163
9	17,951	404
10	9,694	204
Total	84,174	1,828

Territories 2 & 3 don't exist in the Returns table, so they aren't included in the filter context that passes to Territory Lookup and Sales

Hide - Foreign Key

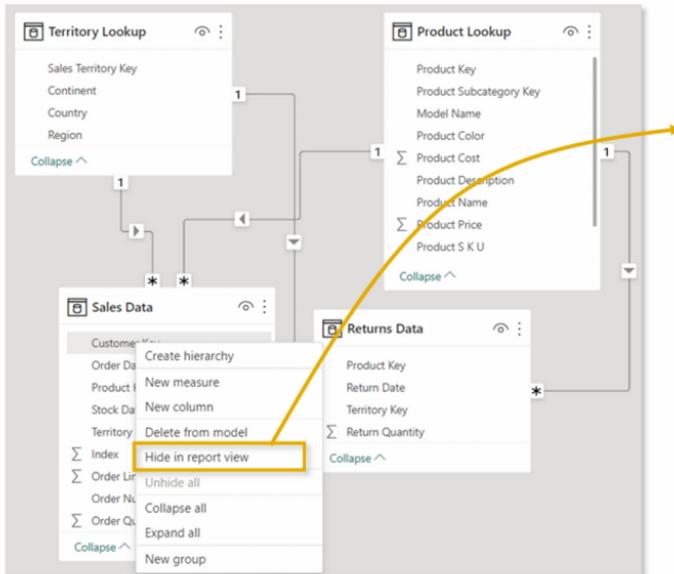
All the F.K present on Fact Table has to be hidden. To avoid wrong insight on Dashboard.



You always get a chance to choose primary key to get the final result.



HIDING FIELDS



Hide in Report View makes fields inaccessible from the Report tab, but still available in Data and Model views

- This can be controlled by right-clicking a field in the Data or Model view, or by selecting "Is hidden" in the Properties pane

- This is commonly used to prevent users from filtering using invalid fields, reduce clutter, or to hide irrelevant metrics from view

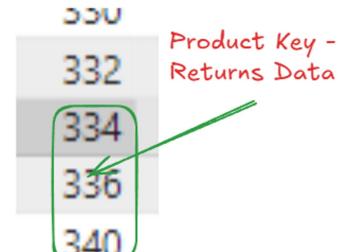
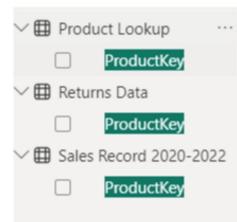
PRO TIP: Hide the foreign keys in fact tables to force users to filter using primary keys in dimension tables

ASSIGNMENT: FILTER FLOW

1. Replicate matrix below to diagnose what he must have done to the model

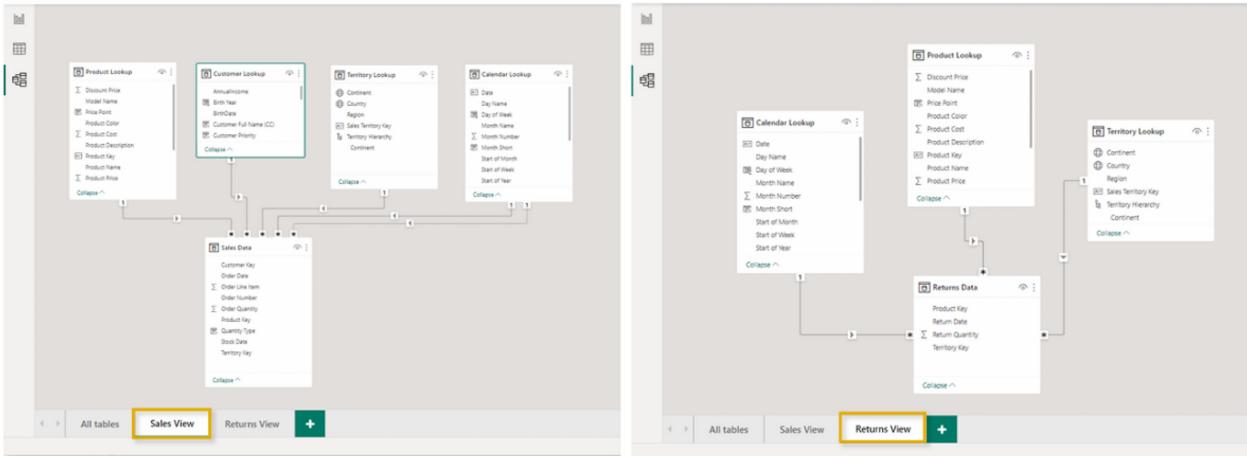
Product Key	Sum of Order Quantity	Sum of Return Quantity
324	72	3
326	65	3
328	75	4
330	51	6
332	64	2
334	63	2
336	50	1
340	56	1
342	72	1
346	24	2

- Which product is #338?
- Why didn't matrix show any orders?



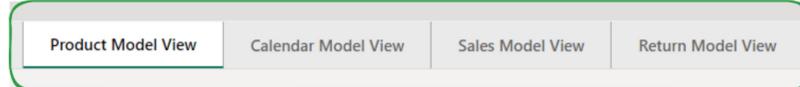
2. Hide any remaining foreign keys to prevent other users from making the same mistake

MODEL LAYOUTS



--> Model layouts allow you to create custom views to show specific portions of large, complex models

- Here we've created a Sales View displaying only tables related to sales, and a Returns View displaying only tables related to returns (Note: this doesn't actually create duplicate tables)



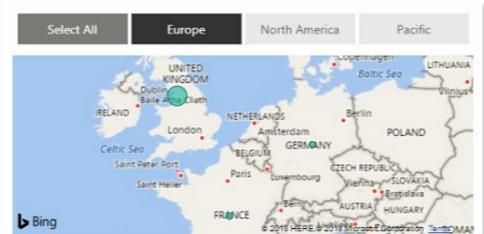
DATA FORMATS & CATEGORIES

Customize data formats from the Column tools menu in the Data view or the Properties pane in the Model view

The screenshot shows the Power BI Data View interface. The ribbon at the top has 'Table tools' selected. In the 'Column tools' section, the 'Formatting' dropdown is highlighted with a yellow box. Below it, the 'Data category' dropdown is also highlighted with a yellow box, showing 'Country' as the selected option. To the right of the dropdown, there's a 'Sort by column' button with a yellow arrow pointing to it.

Assign data categories for geospatial fields, URLs or barcodes

- This is commonly used to help Power BI map location-based fields like addresses, countries, cities, coordinates, zip codes, etc.



HIERARCHIES

Hierarchies are groups of columns that reflect multiple levels of granularity

- For example, a Geography hierarchy might include Country, State and City fields
- Hierarchies are treated as a single item in tables and reports, allowing users to "drill up" and "drill down" through each level

The image consists of three screenshots related to hierarchy creation in Power BI:

- Left Screenshot:** Shows the Power BI Data View with a context menu open over the 'Product Price' field. The 'Create hierarchy' option is highlighted with a yellow box.
- Middle Screenshot:** Shows the Power BI Model View. A hierarchy named 'Territory Hierarchy' is selected. It contains three levels: 'Continent', 'Country', and 'Region'. The 'Sales Territory Key' field is also visible.
- Right Screenshot:** Shows the Power BI Data View with a context menu open over the 'Country' field. The 'Add to hierarchy' option is highlighted with a yellow box.

In the Data pane, right-click a field and select Create hierarchy

This hierarchy contains "Continent", and is named "Territory Hierarchy"

Right-click another field (like "Country") and select Add to Hierarchy (or drag it in!)

ASSIGNMENT: HIERARCHIES

1. Create a new hierarchy based on the Start of Year field, and name it "Date Hierarchy".
2. Right-click or drag to add fields until your hierarchy contains the following (in this order):
 - Start of Year
 - Start of Month
 - Start of Week
 - Date
3. Add your new hierarchy to the matrix visual (on rows) and practice drilling up and down between each level of granularity

DATA MODEL BEST PRACTICES

Focus on building a normalized model from the start

- Leverage relationships and make sure that each table serves a clear, distinct purpose

Organize dimension tables above data tables in your model

- This serves as a visual reminder that filters always flow "downstream"

Avoid complex relationships unless absolutely necessary

- Aim to use 1-to-many table relationships and one-way filters whenever possible

Hide fields from report view to prevent invalid filter context

- This forces report users to filter using primary keys from dimension tables

Territory Lookup

Continent

Advanced

Sort by column: Continent (Default)

Data category: Continent

Territory Lookup

Continent
Country
Region

Name: Continent
Format: Text
Data type: Text

SalesTerritoryKey	Region	Country	Continent
1	Northwest	United States	North America
2	Northeast	United States	North America
3	Central	United States	North America
4	Southwest	United States	North America
5	Southeast	United States	North America
6	Canada	Canada	North America
7	France	France	Europe
8	Germany	Germany	Europe
9	Australia	Australia	Pacific
10	United Kingdom	United Kingdom	Europe

Uncategorized

Address

City

Continent
Country/Region

County

Latitude

Longitude

Place

Postal code

State or Province

Web URL

Image URL

Barcode

Continent

Either do it from Model View, or from Table View. Both will produce the same result.

Date Day Name Start of Week Start of Month Month Name

01-01-2020	Wednesday	29-12-2019	01-01-2020	January
08-01-2020	Wednesday	05-01-2020	01-01-2020	January
15-01-2020	Wednesday	12-01-2020	01-01-2020	January
22-01-2020	Wednesday	19-01-2020	01-01-2020	January
29-01-2020	Wednesday	26-01-2020	01-01-2020	January
05-02-2020	Wednesday	02-02-2020	01-02-2020	February
12-02-2020	Wednesday	09-02-2020	01-02-2020	February
19-02-2020	Wednesday	16-02-2020	01-02-2020	February
26-02-2020	Wednesday	23-02-2020	01-02-2020	February
04-03-2020	Wednesday	01-03-2020	01-03-2020	March
11-03-2020	Wednesday	08-03-2020	01-03-2020	March
18-03-2020	Wednesday	15-03-2020	01-03-2020	March
25-03-2020	Wednesday	22-03-2020	01-03-2020	March
01-04-2020	Wednesday	29-03-2020	01-04-2020	April
08-04-2020	Wednesday	05-04-2020	01-04-2020	April
15-04-2020	Wednesday	12-04-2020	01-04-2020	April
22-04-2020	Wednesday	19-04-2020	01-04-2020	April
29-04-2020	Wednesday	26-04-2020	01-04-2020	April

ProductCost ProductPrice

\$2,171.2942	₹ 3,578.27
\$2,171.2942	₹ 3,578.27
\$2,171.2942	₹ 3,578.27
\$2,171.2942	₹ 3,578.27
\$2,171.2942	₹ 3,578.27
\$2,171.2942	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$884.7083	₹ 1,457.99
\$412.1463	₹ 600.0000

By Default it is Long Date format , Convert it into Short Date.

Format: A
Summarization: Don't summarize

Date formats

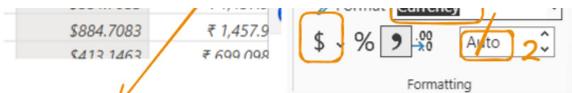
t of Week

14-03-2001 (Short Date)
14 March 2001 (Long Date)
14 March 2001 (dd mmmm yyyy)
14 March 2001 (d mmmm yyyy)
Wednesday, 14 March, 2001 (dddd, d mmmm, yyyy)
14-03-2001 (dd-mm-yyyy)
14-03-01 (dd-mm-yy)
14-3-01 (d-m-yy)
14.3.01 (d.m.y)
2001-03-14 (yyyy-mm-dd)
01-03-2006 (dd/mm/yyyy)
14-03-2001 (dd/mm/yyyy)
14-03-2001 (dd/mm/yyyy)
March, 2001 (mmmm, yyyy)
2001-03 (yyyy-mm)
29-03-2006 (dd/mm/yyyy)
14 March (d mmmm)
05-04-2006 (dd/mm/yyyy)
12-04-2006 (dd/mm/yyyy)
19-04-2006 (dd/mm/yyyy)
26-04-2006 (dd/mm/yyyy)

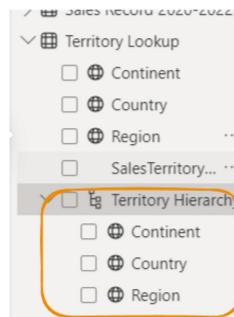
Format: \$
Format: %
Format: ,
Format: .
Format: 2

Formatting

change the currency to \$.



Make it 2 decimal places.



Continent

Country

Region

Hierarchy Flow

Matrix Visual

Continent	Sum of OrderQuantity	Sum of ReturnQuantity
Europe	25506	553
North America	40717	871
Pacific	17951	404
Total	84174	1828

Drill up & Drill Down

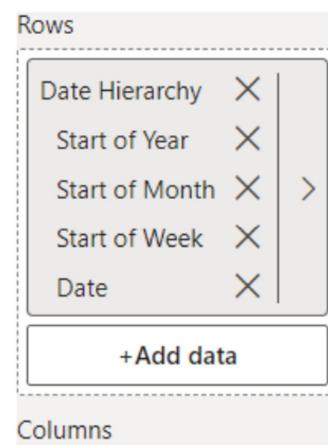
Continent	Sum of OrderQuantity	Sum of ReturnQuantity
France	7862	186
Germany	7950	163
United Kingdom	9694	204
North America	40717	871
Canada	10894	238
United States	29823	633
Central	30	
Northeast	40	
Northwest	12513	270
Southeast	49	1
Southwest	17191	362
Pacific	17951	404
Australia	17951	404
Total	84174	1828

Adding Start of Year helping me to Create Date Hierarchy

Granular Level

Date	Start of Year	Day Name	Start of Week	Start of Month	Month Name
● Valid ○ Error ● Empty	100% 0% 0%	100% 0% 0%	100% 0% 0%	100% 0% 0%	100% 0% 0%
912 distinct, 912 unique	3 distinct, 0 unique	7 distinct, 0 unique	131 distinct, 0 unique	30 distinct, 0 unique	12 distinct, 0 unique
1 01-01-2020	01-01-2020	Wednesday	29-12-2019	01-01-2020	January
2 02-01-2020	01-01-2020	Thursday	29-12-2019	01-01-2020	January
3 03-01-2020	01-01-2020	Friday	29-12-2019	01-01-2020	January
4 04-01-2020	01-01-2020	Saturday	29-12-2019	01-01-2020	January

Start of Year	Sum of OrderQuantity	Sum of ReturnQuantity
01 January 2020	2630	86
01-01-2020	184	4
29-12-2019	21	
05-01-2020	38	
12-01-2020	47	2
19-01-2020	43	2
26-01-2020	35	
01-02-2020	165	4
26-01-2020	5	
02-02-2020	46	1
03-02-2020	45	1



09-02-2020	40	1
16-02-2020	30	1
23-02-2020	38	1
01-03-2020	198	9
01-03-2020	50	
08-03-2020	37	3
15-03-2020	49	1
22-03-2020	48	4
Total	84174	1828

+Add data

Values

Sum of OrderQuantity X

Sum of ReturnQuantity X

+Add data