< Takaisin välilehdelle

Tee: Käy oppitunti läpi loppuun asti

Tasks

In *Lesson 4*, *Building the Data Model*, you learned how to develop your data model to deal with role-playing tables, by importing a table multiple times.

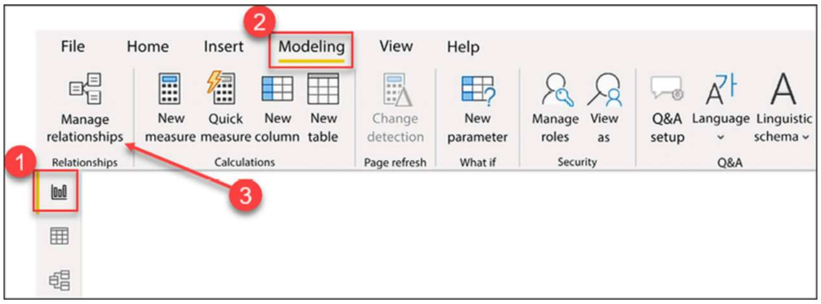
There is an alternative method using DAX. In this exercise , we will explore this alternative method and the pros and cons of using DAX versus the method you have previously learned.

Since leveraging DAX does not require importing a table multiple times, you will immediately gain savings on storage and, unlike the other method, with DAX, you will not need to manage multiple tables in Power BI Desktop.

The DAX method requires that **inactive relationships be created** in order to work correctly. Inactive relationships are not often used in DAX because they are not used automatically like active relationships. Unlike active relationships, you can have more than one inactive relationship between two tables.

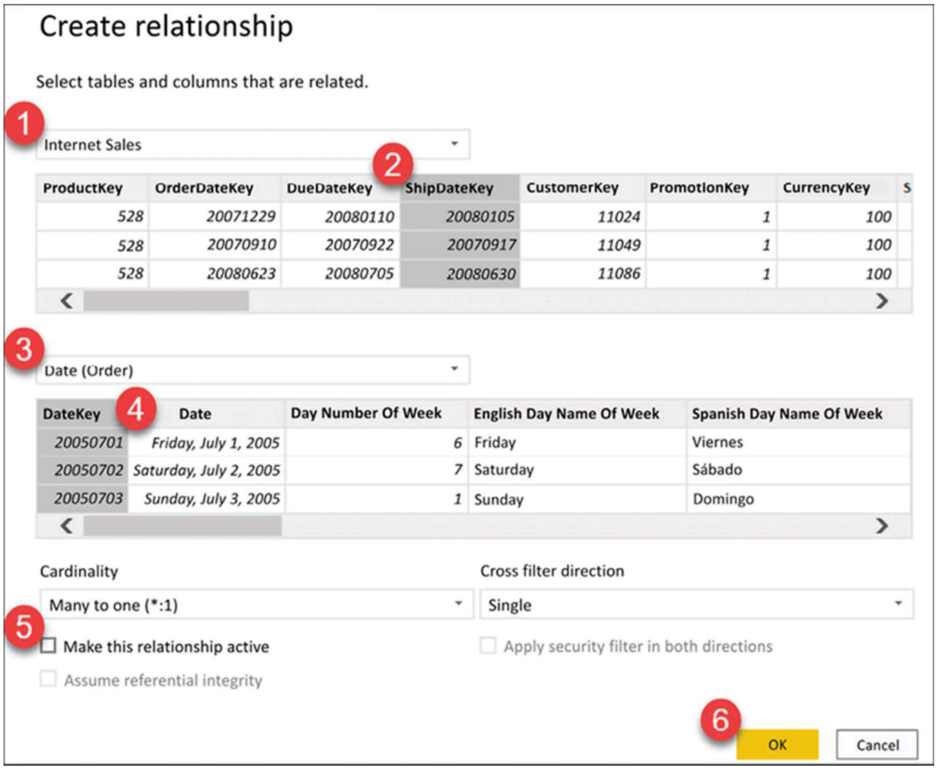
Step 1: Create a new relationship between the *Internet Sales* table and the *Date (Order)* table:

1. Navigate to the *Report View.*
2. Select the *Modeling ribbon* across the top of Power BI Desktop.
3. Click on **Manage relationships.**



Step 2: Click on **New...** to create a new relationship:

1. Select *Internet Sales* from the drop-down menu.
2. Select the *ShipDateKey* column from the list of columns.
3. Select *Date (Order)* from the drop-down menu.
4. Select the *DateKey* column from the list of columns.
5. **Do not select** *Make this relationship active*.
6. Click **OK**.



Create relationship

Select tables and columns that are related.

Internet Sales

ProductKey	OrderDateKey	DueDateKey	ShipDateKey	CustomerKey	PromotionKey	CurrencyKey	S
528	20071229	20080110	20080105	11024	1	100	
528	20070910	20070922	20070917	11049	1	100	
528	20080623	20080705	20080630	11086	1	100	

Date (Order)

DateKey	Date	Day Number Of Week	English Day Name Of Week	Spanish Day Name Of Week
20050701	Friday, July 1, 2005	6	Friday	Viernes
20050702	Saturday, July 2, 2005	7	Saturday	Sábado
20050703	Sunday, July 3, 2005	1	Sunday	Domingo

Cardinality: Many to one (*:1)

Cross filter direction: Single

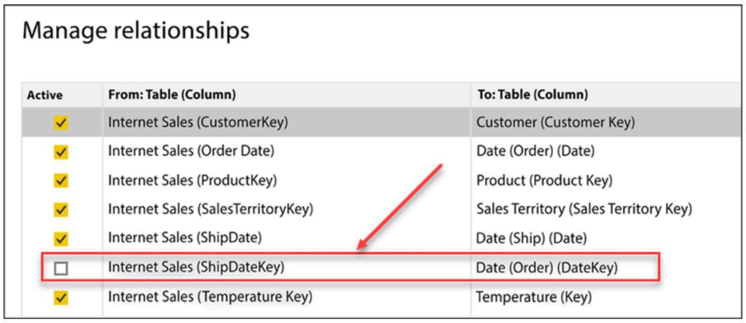
☐ Make this relationship active

☐ Apply security filter in both directions

☐ Assume referential integrity

OK Cancel

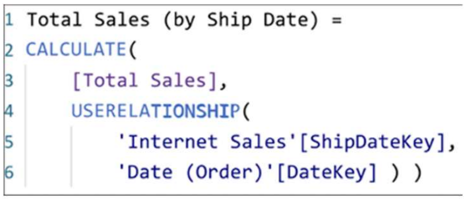
The new relationship can be observed in Figure below, which is a screenshot of the *Manage relationships* dialog box:



Manage relationships

Active	From: Table (Column)	To: Table (Column)
☑	Internet Sales (CustomerKey)	Customer (Customer Key)
☑	Internet Sales (Order Date)	Date (Order) (Date)
☑	Internet Sales (ProductKey)	Product (Product Key)
☑	Internet Sales (SalesTerritoryKey)	Sales Territory (Sales Territory Key)
☑	Internet Sales (ShipDate)	Date (Ship) (Date)
☐	Internet Sales (ShipDateKey)	Date (Order) (DateKey)
☑	Internet Sales (Temperature Key)	Temperature (Key)

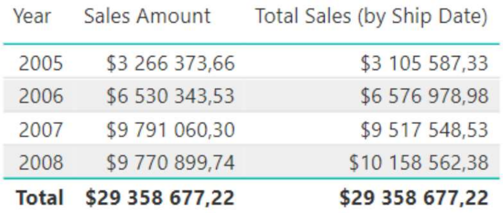
Step 3: Create the calculated measure to return **Total Sales by Ship Date**.:



```
Total Sales (by Ship Date) =
CALCULATE(
    [Total Sales],
    USERELATIONSHIP(
        'Internet Sales'[ShipDateKey],
        'Date (Order)'[DateKey] ) )
```

This measure will make use of two functions in DAX. First, the CALCULATE function is used here because the filter context is going to be modified to use the inactive relationship rather than the active relationship. Second, the USERELATIONSHIP function specifies that the Internet Sales table should be filtered by ShipDateKey rather than the active relationship on OrderDateKey.

The completed measure can be seen in figure below:



Year	Sales Amount	Total Sales (by Ship Date)
2005	$3 266 373,66	$3 105 587,33
2006	$6 530 343,53	$6 576 978,98
2007	$9 791 060,30	$9 517 548,53
2008	$9 770 899,74	$10 158 562,38
Total	**$29 358 677,22**	**$29 358 677,22**

In Figure, the Total Sales measure is returning the total sales based on the active relationship in the data model, which is on *OrderDateKey*, therefore $3 266 373 is returned for the year 2005. Alternatively, the *Total Sales (by Ship Date)* measure is returning $3 105,587.33 in sales.

This approach does not require importing additional tables and therefore is superior to the method you learned in *Lesson 4*, *Building the Data Model*, for optimizing space in your data model. However, with the DAX method, you would be required to create a new measure for every measure in your data model that you wanted to see by the Ship Date. Therefore, if you had 50 measures in your table, you would create a new version of each of those measures and would specify that the new measure should use the inactive relationship on *ShipDateKey* rather than the current active relationship. Because of this reason alone, the method you learned in *Lesson 4*, *Building the Data Model*, is the most common approach to handling role-playing tables.

With the addition of external tools and calculation groups, you can now create one measure to support role-playing tables! This makes the DAX method significantly easier to implement than ever before. Want to learn more? Check out the following YouTube video on this implementation: https://tinyurl.com/RolePlayingTables.

Note! The *Lesson 5 - Leveraging DAX - Completed.pbix* file can be downloaded from this **link**.

End-of-Exercise

Olet suorittanut 0 % oppitunnista

◄ Exercise 14 - Working with time intelligence functions

Siirry...

Lesson 5 Quiz ►