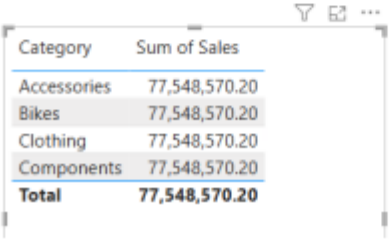
1. Create model relationships
2. Create additional relationships
3. Configuring each tables
4. Arrive with sum of sales and Sum of Targets with each salesperson

Create model relationships

In this task, you'll create model relationships. The file was configured to not identify relationships between tables, which isn't the default setting, but is recommended to prevent extra work creating the correct relationships for your model.

The labs use a shorthand notation to reference a field. It will look like this: `Product | Category`. In this example, `Product` is the table name and `Category` is the field name.

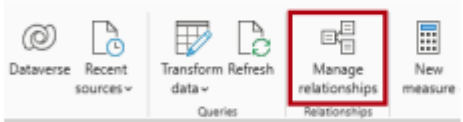
1. In Power BI Desktop, to view all table fields, in the Data pane, right-click an empty area, and then select Expand All.
2. To create a table visual, in the Data pane, from inside the `Product` table, check the `Category` field.
3. To add another column to the table, in the Data pane, check the `Sales | Sales` field.
4. Notice that the table visual lists four product categories, and that the sales value is the same for each, and the same for the total.



Category	Sum of Sales
Accessories	77,548,570.20
Bikes	77,548,570.20
Clothing	77,548,570.20
Components	77,548,570.20
Total	**77,548,570.20**

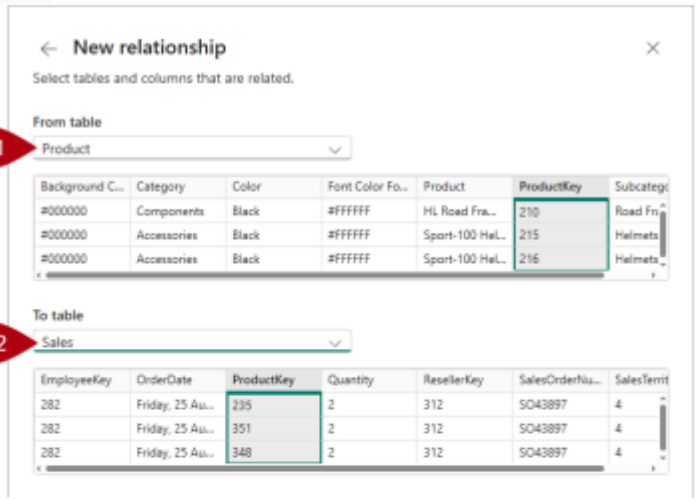
5. *The issue is that the table is based on fields from different tables. The expectation is that each product category displays the sales for that category. However, because there isn't a model relationship between these tables, the `Sales` table isn't filtered. You'll now add a relationship to propagate filters between the tables.*
6. To switch to the model designer, at the left select the Model view icon.

7. On the Home ribbon, select Manage Relationships.



8. In the Manage Relationships window, notice that no relationships are yet defined.
9. To create a relationship, select + New relationship.
10. To configure a relationship from `Product` table to `Sales` table, in the From table dropdown list, select the `Product` table, and in the To table dropdown list, select

the `Sales` table.



11. Notice the following properties were automatically configured:
    - ProductKey columns in each table are selected. *The columns were selected because they share the same name and data type. You may need to find matching columns with different names in real data.*
    - Cardinality type is One To Many (1:*). *The cardinality was automatically detected, because Power BI understands that the `ProductKey` column from the `Product` table contains unique values. One-to-many relationships are the most common cardinality, and all relationship you create in this lab will be this type.*
    - Cross Filter Direction type is Single. *Single filter direction means that filters propagate from the "one side" to the "many side". In this case, it means filters applied to the `Product` table will propagate to the `Sales` table, but not in the opposite direction.*
    - Make This Relationship Active is checked. *Active relationships propagate filters. It's possible to mark a relationship as inactive so filters don't propagate. Inactive relationships can exist when there are multiple relationship paths between tables. In this case, model calculations can use special functions to activate them.*

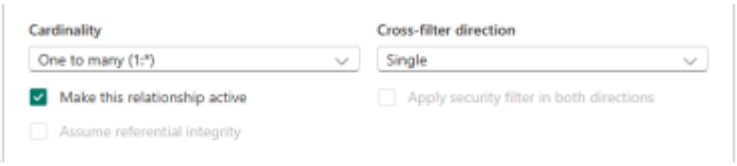
19.

Category	Sum of Sales
Accessories	539,106.09
Bikes	64,069,033.16
Clothing	1,714,056.05
Components	11,226,374.90
Total	**77,548,570.20**

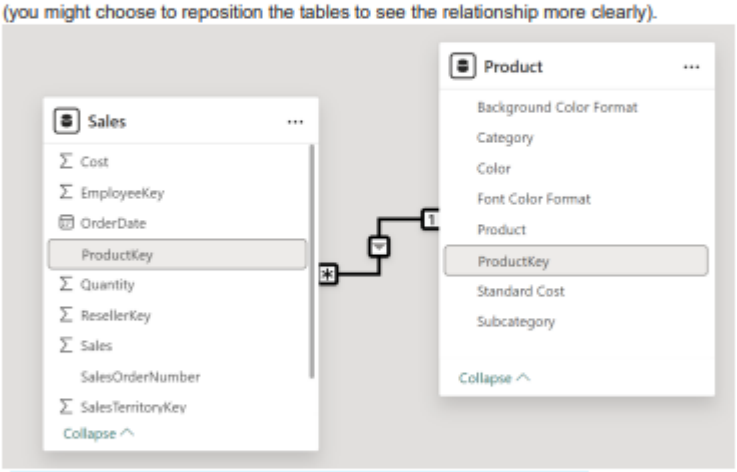
Create additional relationships

There's an easier way to create a relationship. In the model diagram, you can drag and drop columns to create a new relationship.

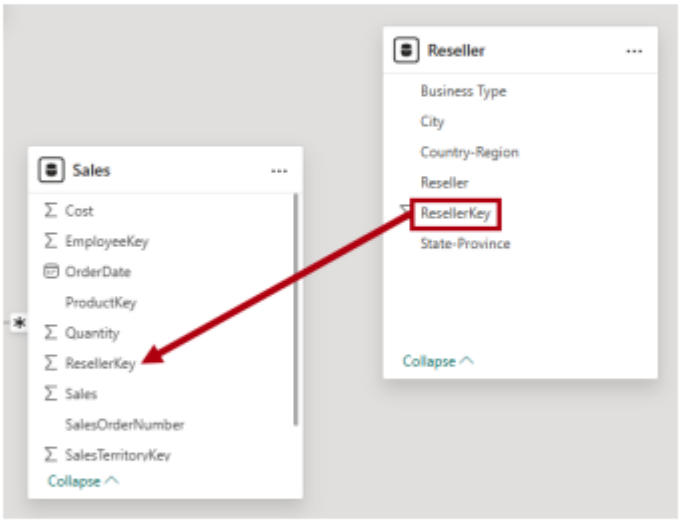
1. To create a new relationship using a different technique, switch to Model view.
2. From the `Reseller` table, drag the `ResellerKey` column onto the `ResellerKey` column of the `Sales` table.
3. *Important: Sometime a column doesn't want to be dragged. If this situation arises, select a different column, and then select the column you intend to drag again, and then try again. Ensure that you see the new relationship added to the diagram.*



12.
13. Select Save, notice in the Manage Relationships window that the new relationship is listed, and then select Close.
14. In the model diagram, notice there's now a connector between the two tables (you might choose to reposition the tables to see the relationship more clearly).

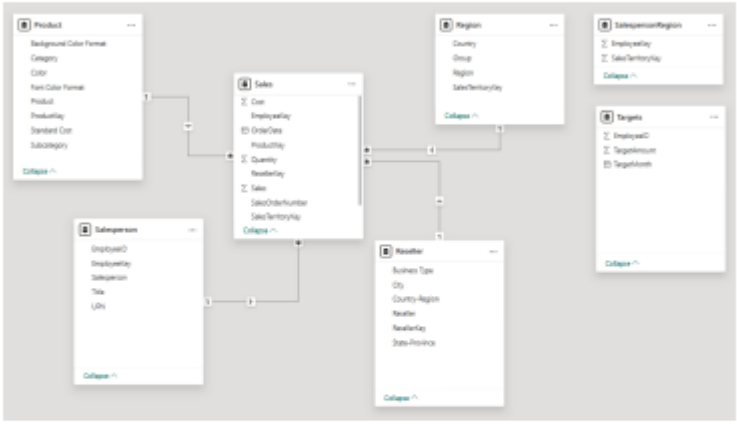


15. *You can interpret many of the relationship properties by looking at the relationship line in the diagram:*
    - *Cardinality is represented by the 1 and (*) indicators.*
    - *Filter direction is represented by the arrow head.*
    - *A solid line represents an active relationship; a dotted line represents an inactive relationship.*
16. Tip: If you hover the cursor over the relationship to highlight the related columns._
17. Switch to Report view, and then notice that the table visual updated to display different values for each product category.
18. *Filters applied to the `Product` table now propagate to the `Sales` table.*



4.
5. In the New relationship window, review the configuration, and then select Save.
6. Use the new technique to create the following two model relationships:
   - `Region | SalesTerritoryKey` to `Sales | SalesTerritoryKey`
   - `Salesperson | EmployeeKey` to `Sales | EmployeeKey`
7. In the diagram, arrange the tables so that the `Sales` table is positioned in the center of the diagram, and the related tables are arranged about it. Position the

disconnected tables to the side.



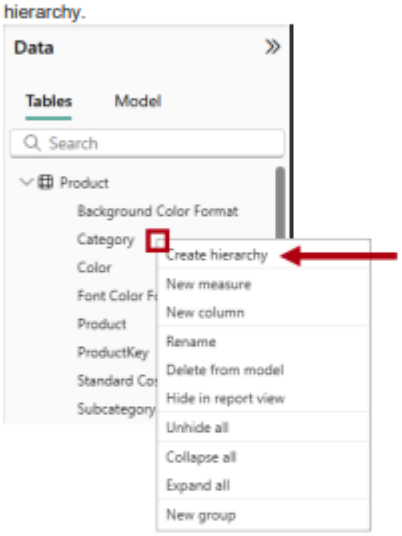
8. Save the Power BI Desktop file.

Configure the Product table

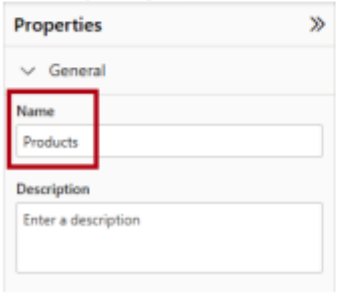
In this task, you'll configure the `Product` table with a hierarchy and display folder.

1. Switch to Model view.
2. In the Data pane, if necessary, expand the `Product` table to reveal all fields.

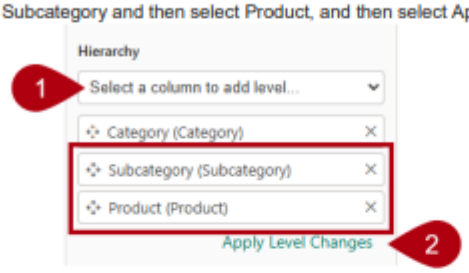
3. In the `Product` table, right-click the `Category` column, and the select Create hierarchy.



4. In the Properties pane, in the Name box, replace the value with *Products*.

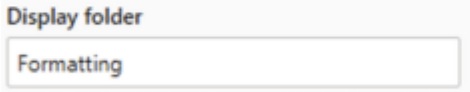


5. Two add levels to the hierarchy, in the Hierarchy dropdown list, select Subcategory and then select Product, and then select Apply Level Changes.

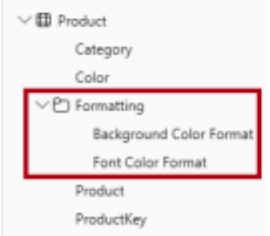


6. In the Data pane, notice the `Products` hierarchy. To reveal the hierarchy levels, expand it.

7. To organize columns into a display folder, in the Data pane, first select the `Background Color Format` column.
8. While pressing the Ctrl key, select the `Font Color Format` column.
9. In the Properties pane, in the Display Folder box, enter *Formatting*.



10. In the Data pane, notice that the two columns are now inside a folder.

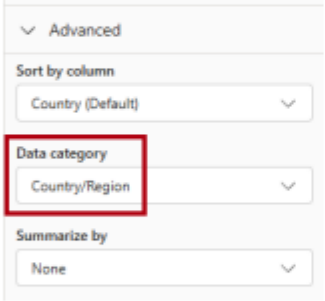


11. *Display folders are a great way to organize tables, especially for tables that comprise many fields. They're logical presentation only.*

Configure the Region table

In this task, you'll configure the `Region` table with a hierarchy and updated categories.

1. In the `Region` table, create a hierarchy named *Regions*, with the following three levels:
   - `Group`
   - `Country`
   - `Region`
2. Select the `Country` column (not the `Country` hierarchy level).
3. In the Properties pane, expand the Advanced section (located at the bottom of the pane), and then in the Data Category dropdown list, select Country/Region.



4. *Data categorization can provide hints to the report designer. In this case, categorizing the column as country or region provides more accurate information to Power BI when it renders a map visualization.*

Configure the Reseller table

In this task, you'll configure the `Reseller` table to add a hierarchy and update data categories.

1. In the `Reseller` table, create a hierarchy named *Resellers*, with the following two levels:
   - `Business Type`
   - `Reseller`
2. Create a second hierarchy named *Geography*, with the following four levels:
   - `Country-Region`

    - `State-Province`
    - `City`
    - `Reseller`
3. Set the data category for the following columns (not within the hierarchy):
    - `Country-Region` to Country/Region
    - `State-Province` to State or Province
    - `City` to City

Configure the Sales table

In this task, you'll configure the `Sales` table with updated descriptions, formatting, and summarization.

1. In the `Sales` table, select the `Cost` column.
2. In the Properties pane, in the Description box, enter: *Based on standard cost*
3. *Descriptions can be applied to tables, columns, hierarchies, or measures. In the Data pane, description text is revealed in a tooltip when a report author hovers their cursor over the field.*
4. Select the `Quantity` column.
5. In the Properties pane, from inside the Formatting section, set the Thousands Separator property to *Yes*.
6. Select the `Unit Price` column.
7. In the Properties pane, from inside the Formatting section, set the Decimal Places property to *2*.
8. In the Advanced group (you may need to scroll down to locate it), in the Summarize By dropdown list, select Average.
9. *By default, numeric columns will summarize by summing values together. This default behavior isn't suitable for a column like `Unit Price`, which represents a rate. Setting the default summarization to average will produce a meaningful result.*

Bulk update properties

In this task, you'll update multiple columns using single bulk updates. You'll use this approach to hide columns, and format column values.

1. In the Data pane (or model diagram), select the `Product | ProductKey` column.

2. While pressing the Ctrl key, select the following 13 columns (spanning multiple tables):
    - `Region | SalesTerritoryKey`
    - `Reseller | ResellerKey`
    - `Sales | EmployeeKey`
    - `Sales | ProductKey`
    - `Sales | ResellerKey`
    - `Sales | SalesOrderNumber`
    - `Sales | SalesTerritoryKey`
    - `Salesperson | EmployeeID`
    - `Salesperson | EmployeeKey`
    - `Salesperson | UPN`
    - `SalespersonRegion | EmployeeKey`
    - `SalespersonRegion | SalesTerritoryKey`
    - `Targets | EmployeeID`
3. In the Properties pane, set the Is Hidden property to *Yes*.
4. *The columns were hidden because they're either used by relationships or will be used in row-level security configuration or calculation logic.*
5. *You'll use the `SalesOrderNumber` column in a calculation in the Create DAX Calculations in Power BI Desktop lab.*
6. Multi-select the following three columns:
    - `Product | Standard Cost`
    - `Sales | Cost`
    - `Sales | Sales`
7. In the Properties pane, from inside the Formatting section, set the Decimal Places property to *0* (zero).

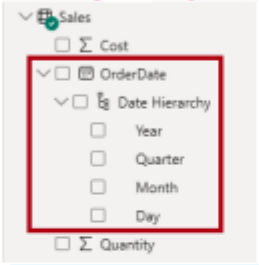
Explore the model interface

In this task you'll switch to Report view, review the data model interface, and configure the auto date/time setting.

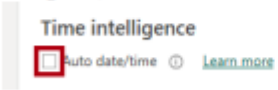
1. Switch to Report view.
2. In the Data pane, notice the following:
    - Columns, hierarchies and their levels are fields, which can be used to configure report visuals.
    - Only fields relevant to report authoring are visible.
    - The `SalespersonRegion` table isn't visible—because all of its fields are hidden.

- Spatial fields in the `Region` and `Reseller` table are adorned with a spatial icon.
- Fields adorned with the sigma symbol (Σ) will summarize, by default.
- A tooltip appears when hovering the cursor over the `Sales | Cost` field.

3. Expand the `Sales | OrderDate` field, and then notice that it reveals a `Date Hierarchy`. The `Targets | TargetMonth` field delivers a similar hierarchy.



4. *Important: These hierarchies weren't created by you. They were created automatically as a default setting. There's a problem, however. The Adventure Works financial year commences on July 1 of each year. But, in these automatically created date hierarchies, the date hierarchy year commences on January 1 of each year.*
5. To turn off the auto date/time setting, navigate to File > Options and Settings > Options.
6. In the Options window, on the Current File section, navigate to Data Load > Time Intelligence, and uncheck Auto Date/Time.



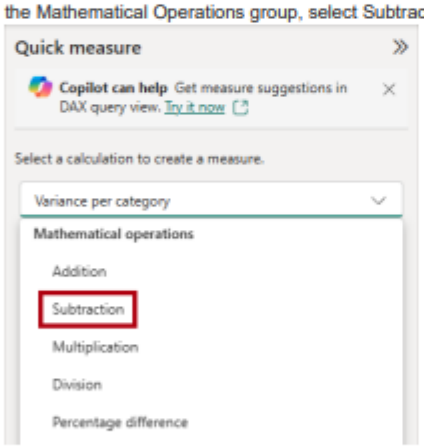
7. In the Data pane, notice that the date hierarchies are no longer available.

Create quick measures

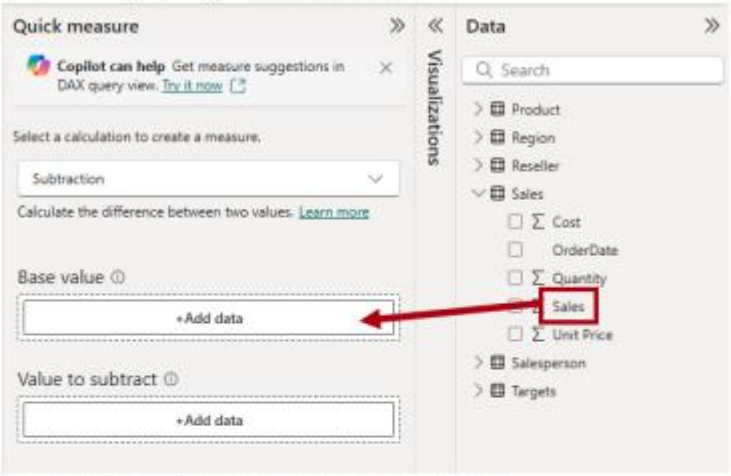
In this task, you'll create two quick measures to calculate profit and profit margin. A quick measure creates the calculation formula for you. They're easy and fast to create for simple and common calculations.

1. In the Data pane, right-click the `Sales` table, and then select New Quick

2. In the Quick Measure pane, in the Select a Calculation dropdown list, from inside the Mathematical Operations group, select Subtraction.

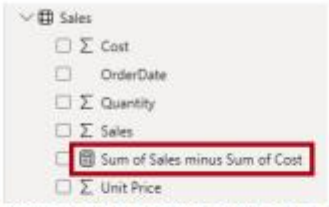


3. From the Data pane, drag the `Sales | Sales` field into the Base Value well.



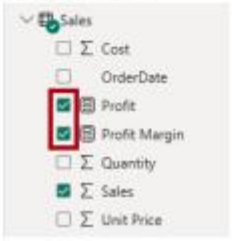
4. Drag the `Sales | Cost` field into the Value to Subtract box.

5. Select Add.
6. In the Data pane, inside the `Sales` table, notice that new measure.
7. *Measures are indicated by the calculator icon.*



8.
9. To rename the measure, right-click it, select Rename, then rename as *Profit*.
10. *Tip: To rename a field, you can also double-click it, or select it and press F2.*
11. In the `Sales` table, add a second quick measure, based on the following requirements:
12. _Important: If the option to create a quick measure doesn't appear in the context menu, use the command located on the Home ribbon, from inside the Calculations group.
    - Use the Division mathematical operation.
    - Set the Numerator to the `Sales | Profit` field.
    - Set the Denominator to `Sales | Sales` field.
    - Rename the measure as *Profit Margin*.
13. Ensure the `Profit Margin` measure is selected, and then on the Measure Tools contextual ribbon, set the format to Percentage, with two decimal places.

14. To test the two measures, first select the existing table visual on the page.
15. In the Data pane, check the `Profit` and `Profit Margin` measures to add them to the table.



16. Select and drag the right guide to widen the table visual.

Category	Sales	Profit	Pro
Accessories	$539,106	$188,081	
Bikes	$64,069,033	($435,679)	
Clothing	$1,714,056	$245,857	
Components	$11,226,375	$1,001,235	
Total	**$77,548,570**	**$999,495**	

17. Verify that the measures produce reasonable results that are correctly formatted.

Category	Sales	Profit	Profit Margin
Accessories	$539,106	$188,081	34.89 %
Bikes	$64,069,033	($435,679)	-0.68 %
Clothing	$1,714,056	$245,857	14.34 %
Components	$11,226,375	$1,001,235	8.92 %
Total	**$77,548,570**	**$999,495**	**1.29 %**

Create a many-to-many relationship

In this task, you'll create a many-to-many relationship between the `Salesperson` table and the `Sales` table.

1. In Report view, select a blank area of the report page.
2. To create a new table visual, in the Data pane, check the following two fields:
   - `Salesperson | Salesperson`
   - `Sales | Sales`

Salesperson	Sum of Sales
Amy Alberts	$737,568
David Campbell	$3,614,761
Garrett Vargas	$3,486,102
Jae Pak	$8,099,817
Jillian Carson	$9,755,992
José Saraiva	$5,536,439
Linda Mitchell	$10,158,635
Lynn Tsoflias	$1,216,917
Michael Blythe	$8,952,751
Pamela Ansman-Wolfe	$3,187,720
Rachel Valdez	$1,681,208
Ranjit Varkey Chudukatil	$4,429,368
Shu Ito	$6,283,193
Stephen Jiang	$1,073,651
Syed Abbas	$174,108
Tete Mensa-Annan	$2,184,211
Tsvi Reiter	$6,976,128
Total	**$77,548,570**

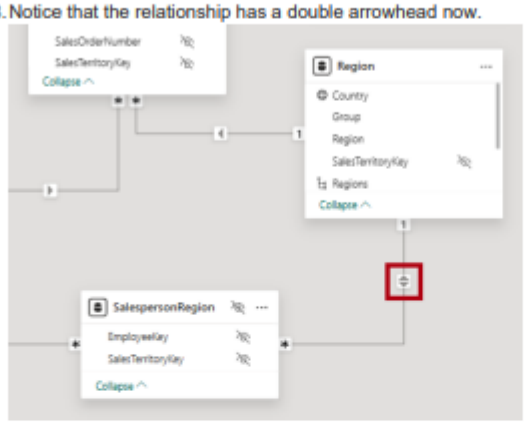
3.
4. *The table visual displays sales made by each salesperson. However, there's another relationship between salespeople and sales. Some salespeople belong to one, two, or possibly more sales regions. In addition, sales regions can have multiple salespeople assigned to them.*
5. *From a performance management perspective, a salesperson's sales (based on their assigned territories) need to be analyzed and compared with sales targets. You'll create relationships to support this analysis in the next exercise.*
6. Notice that *Michael Blythe* has generated almost 9 million dollars of sales.
7. Switch to Model view, then drag the `SalespersonRegion` table to position it between the `Region` and `Salesperson` tables.
8. Use the drag-and-drop technique to create the following two model relationships:
   - `Salesperson | EmployeeKey` to `SalespersonRegion | EmployeeKey`
   - `Region | SalesTerritoryKey` to `SalespersonRegion | SalesTerritoryKey`
9. *The `SalespersonRegion` table can be considered to be a bridging table.*
10. Switch to Report view, and then notice that the visual hasn't updated—the sales result for Michael Blythe hasn't changed.
11. Switch back to Model view, and then follow the relationship filter directions (arrowhead) from the `Salesperson` table.

12. *Consider that the `Salesperson` table filters the `Sales` table. It also filters the `SalespersonRegion` table, but it doesn't continue by propagating filters to the `Region` table (the arrowhead is pointing the wrong direction).*

13.
14. To edit the relationship between the `Region` and `SalespersonRegion` tables, double-click the relationship.
15. In the Edit Relationship window, in the Cross Filter Direction dropdown list, select *Both*.
16. Check the Apply Security Filter in Both Directions checkbox.

17. Select Save.

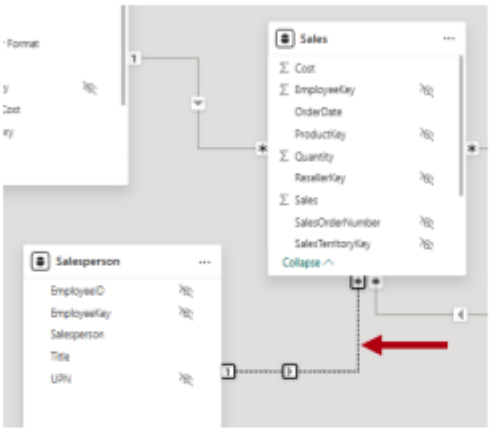
18. Notice that the relationship has a double arrowhead now.



19. Switch to Report view, and then notice that the sales values have still not changed.
20. *The issue now relates to the fact that there are two possible filter propagation paths between the `Salesperson` and `Sales` tables. This ambiguity is internally resolved, based on a "least number of tables" assessment. To be clear, you shouldn't design models with this type of ambiguity—the issue will be addressed in part later in this lab, and by the completion of the Create DAX Calculations in Power BI Desktop lab.*
21. Switch to Model view.
22. To force filter propagation via the bridging table, edit (double-click) the relationship between the `Salesperson` and `Sales` tables.
23. In the Edit Relationship window, uncheck the Make This Relationship Active checkbox.

24. Select Save.
25. *Filter propagation will now follow the only active path.*

26. In the model diagram, notice that the inactive relationship is represented by a dotted line.



27. Switch to Report view, and then notice that the sales for Michael Blythe are now nearly 22 million dollars.
28. Notice also, that the sales for each salesperson—if added—would exceed the table total.
29. *It's a common observation of a many-to-many relationship due to the double, triple, etc. counting of regional sales results. Consider Brian Welcker, the second salesperson listed. His sales amount equals the total sales amount. It's the correct result due to the fact that he's the Director of Sales; his sales are measured by the sales of all regions.*
30. *While the many-to-many relationship is now working, it's now not possible to analyze sales made by a salesperson (because the relationship is inactive). You'll be able to reactivate the relationship when you introduce a calculated table that will allow analyzing sales made in the sales region(s) assigned to the salesperson (for performance analysis) in the Create DAX Calculations in Power BI Desktop lab.*
31. Switch to Model view, and then in the model diagram, select the `Salesperson` table.
32. In the Properties pane, in the Name box, replace the text with *Salesperson (Performance)*.
33. *The renamed table now reflects its purpose: it's used to report and analyze the performance of salespeople based on the sales of their assigned sales regions.*

Relate the Targets table

In this task, you'll create a relationship to the `Targets` table.

1. Create a relationship from the `Salesperson (Performance) | EmployeeID` column and the `Targets | EmployeeID` column.
2. In Report view, add the `Targets | Target` field to the table visual.
3. Resize the table visual so all columns are visible.

Salesperson	Sum of Sales	Sum of Target
Amy Alberts	$10,288,626	$19,450,000
Brian Welcker	$77,548,570	$221,700,000
David Campbell	$12,004,822	$19,625,000
Garrett Vargas	$13,875,633	$23,675,000
Jae Pak	$8,410,883	$13,575,000
Jillian Carson	$7,633,387	$13,675,000
José Saraiva	$13,875,633	$18,875,000
Linda Mitchell	$25,634,503	$40,850,000
Lynn Tsoflias	$1,391,025	$3,210,000
Michael Blythe	$21,987,348	$31,150,000
Pamela Ansman-Wolfe	$30,005,939	$53,850,000
Rachel Valdez	$1,877,743	$4,125,000
Ranjit Varkey Chudukatil	$4,527,840	$9,050,000
Shu Ito	$18,001,116	$59,850,000
Stephen Jiang	$65,868,919	$110,150,000
Syed Abbas	$1,391,025	$3,050,000
Tete Mensa-Annan	$12,004,822	$17,100,000
Tsvi Reiter	$7,638,607	$13,250,000
Total	**$77,548,570**	**$676,210,000**

It's now possible to visualize sales and targets—but for now take care for two reasons. First, there's no filter on a time period, and so targets also include future target amounts. Second, targets aren't additive, and so the total shouldn't be displayed. They can either be disabled by formatting the visual or removed by using calculation logic.

1. Save the Power BI Desktop file.

Lab complete

You may choose to save your Power BI report, though it's not necessary for this lab. In the next exercise, you'll work with a pre-made starter file.