**Lab 4-01: Design a Data Model in Power BI**

**Pre-requisites**

* Download and install Power BI from the Microsoft Store
* The lab links to a localhost SQL Server instance. Download a free Developer copy of the install media.

https://www.microsoft.com/sql-server/sql-server-downloads?SilentAuth=1&f=255&MSPPError=-2147217396&rtc=1

* Install SQL Server from Installation Wizard (Setup)

<https://learn.microsoft.com/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup>

* Install the latest version of Microsoft Edge to access Power BI service online
* Extract the ‘AllFiles’ folder to F:/ and rename it to ‘F:\Allfiles'

<https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/AllfilesDownload.zip>

**Introduction**

In today's data-driven world, organizations are flooded with vast amounts of data from various sources. Power BI is a tool that allows businesses to change this raw data into meaningful insights through visualizations and reports. One of the foundational steps in leveraging Power BI's capabilities is designing an effective data model. A well-structured data model not only ensures accurate and efficient data analysis but also facilitates seamless integration with other data sources and enhances performance.

**Problem**

Organizations often face challenges when dealing with disparate data sources, leading to inconsistent and inaccurate reporting. Without a cohesive data model, analysts struggle to derive insights, resulting in delayed decision-making processes. Common issues include data redundancy, lack of standardized metrics, and inefficient data retrieval, which can hinder a company’s ability to reply quickly to market changes.

**Solution**

To address the problem of fragmented and inefficient data analysis, this lab will guide you through the process of designing a data model in Power BI. You will begin by importing data from various sources, such as Excel spreadsheets, SQL databases, and cloud-based data services. Next, you will explore how to clean and transform this data using Power Query to ensure consistency and accuracy.

Open the 03-Starter-Sales Analysis.pbix file.

***Task 1: Create Model Relationships***

|  |
| --- |
| 1. In Power BI Desktop, at the left, select the **Report view** icon.   Screenshot of the Report icon.   1. To create a table visual, go to the Data pane and select the Category field from the Product table. To add another column to the table, select the Sales field from the Sales table in the Data pane.      1. Notice that the table visual displays four product categories, with the sales value being identical for each category and the overall total.      1. Choose the Model view icon from the left navigation pane, then click on Manage Relationships.   Screenshot of Model icon.   1. In the **Manage Relationships** window, notice that no relationships are yet defined. To create a relationship, select **New relationship**.      1. Set up the relationship from the Product table to the Sales table. Click OK, and in the Manage Relationships window, you will see the new relationship listed. Then, click Close.      1. To create a new relationship using a different method, drag the ResellerKey column from the Reseller table onto the ResellerKey column in the Sales table.      1. Use the new method to establish the following two model relationships:  * Connect Region | SalesTerritoryKey to Sales | SalesTerritoryKey. * Connect Salesperson | EmployeeKey to Sales | EmployeeKey.      1. In the diagram, position the Sales table in the center with the related tables arranged around it. Place the disconnected tables off to the side.      1. In the report view, notice that the table is updated.      1. **Save** the Power BI Desktop file. |

***Task 2: Configure the Product Table***

|  |
| --- |
| 1. In **Model view > Data pane**, if necessary, expand the **Product** table to reveal fields.      1. To create a hierarchy, in the Data pane, right-click the **Category** column and then select **Create Hierarchy**.      1. Update name to **Products** (right-click or double-click to rename). 2. To add a second level to the hierarchy, in the **Properties** pane, in the **Hierarchy** dropdown list, select **Subcategory.**      1. To add a third level to the hierarchy, in the **Hierarchy** dropdown list, select **Product**.      1. To complete the hierarchy design, select **Apply Level Changes**.   Picture 343   1. In the **Data** pane, notice the **Products** hierarchy. To reveal the hierarchy levels, expand the **Products** hierarchy.   Picture 346   1. To organize columns into a display folder, first select the Background Color Format column in the Data pane. While holding the Ctrl key, also select the Font Color Format column.      1. In the **Properties** pane, in the **Display Folder** box, enter **Formatting**.   Picture 348   1. In the **Data** pane, notice that the two columns are now inside a folder.   Picture 349 |

***Task 3: Configure the Region Table***

|  |
| --- |
| 1. In the **Region** table, create a hierarchy named **Regions** with the following three levels:  * Group * Country * Region      1. Click on Country column (not Country hierarchy level). 2. In the Properties pane, expand the Advanced section at the bottom, then select Country/Region from the Data Category. |

***Task 4: Configure Reseller Table***

|  |
| --- |
| 1. In the **Reseller** table, create a hierarchy named **Resellers** with the following levels:      1. Create a second hierarchy named **Geography**, with the following four levels:      1. Set **Data Category** for the following columns (not within the hierarchy):  * Country-Region to **Country/Region** * State-Province to **State or Province** * City\*\* to **City** |

***Task 5: Configure the Sales Table***

|  |
| --- |
| 1. In the Sales table, select the Quantity column. In the Properties pane, go to the Formatting section and set the Thousands Separator property to Yes.      1. Select the Unit Price column. In the Properties pane, navigate to the Formatting section and set the Decimal Places property to 2.      1. In the Advanced group, choose Average from the Summarization dropdown list. |

***Task 6: Bulk Update Properties***

|  |
| --- |
| 1. From **Model view** > **Data** pane, select **Product | ProductKey** column.      1. While pressing the **Ctrl** key, select the following 13 columns:  * Region | SalesTerritoryKey * Reseller | ResellerKey * Targets | EmployeeID      1. In the **Properties** pane, slide the **Is Hidden** property to **Yes**.      1. Multi-select the following three columns:  * Product | Standard Cost * Sales | Cost * Sales | Sales  1. In the Properties pane, within the Formatting section, set the Decimal Places property to 0. |

***Task 7: Explore the Model Interface***

|  |
| --- |
| 1. Switch to **Report** view. Expand the Sales | OrderDate field to reveal a Date Hierarchy. Similarly, the Targets | TargetMonth field provides a comparable hierarchy.   Picture 359   1. To disable the Auto date/time setting, go to File > Options and Settings > Options.      1. In the Current File section, go to Data Load > Time Intelligence and uncheck the Auto Date/Time option.   Picture 362   1. In the **Data** pane, notice that date hierarchies are no longer available. |

***Task 8: Create Quick Measures***

|  |
| --- |
| 1. In the **Data** pane, right-click the **Sales** table and then select **New Quick Measure**.   Picture 366   1. In the **Quick Measures** window, in the **Calculation** dropdown list, from inside the **Mathematical Operations** group, click on **Subtraction**.   Picture 367   1. In the **Data** pane of the **Quick Measures** window, expand the **Sales** table. Drag the **Sales** field into the **Base Value** box.      1. Drag the **Cost** field into the **Value to Subtract** box, then click on **Add**.   Picture 368   1. In the **Data** pane, inside the **Sales** table, notice that new measure.   Picture 370   1. To rename measure, right-click it, select **Rename**, and then rename it to **Profit**. 2. In the **Sales** table, add a second quick measure based on the following requirements:  * Use **Division** mathematical operation * Set **Numerator** to the **Sales | Profit** field * Set **Denominator** to **Sales | Sales** field * Change the name of the measure to **Profit Margin**      1. Ensure the **Profit Margin** measure is selected, and then on the **Measure Tools** contextual ribbon, set the format to **Percentage**, with two decimal places.   Picture 374   1. To test the two measures, start by selecting the Table visual. 2. In the Data pane, check both measures.   Picture 375   1. Ensure that the measures yield accurate results and are formatted correctly.   Picture 378 |

***Task 9: Create a Many-To-Many Relationship***

|  |
| --- |
| 1. In Power BI Desktop, in Report view, in the **Data** pane, check the following two fields to create a new **table** visual.  * Salesperson | Salesperson * Sales | Sales   Table with Salesperson and Sales columns.   1. Observe that Michael Blythe has sold nearly $9 million. 2. Switch to Model view, then drag the SalespersonRegion table to place it between Region and Salesperson tables.      1. Use the drag-and-drop method to create the following two model relationships:  * Salesperson | EmployeeKey to SalespersonRegion | EmployeeKey * Region | SalesTerritoryKey to SalespersonRegion | SalesTerritoryKey      1. Switch to Report view, and then notice that the visual has not been updated—the sales result for Michael Blythe has not changed. 2. Switch to Model view and follow relationship filter directions (arrowhead) originating from the **Salesperson** table.      1. To edit the relationship between **Region** and **SalespersonRegion** tables, double-click the relationship.      1. In the **Edit Relationship** window, in the **Cross Filter Direction** dropdown list, select **Both**. Check **Apply Security Filter in Both Directions** checkbox, then select OK.   Picture 381   1. Notice that the relationship has a double arrowhead.   Picture 382   1. Switch to Report view, and then notice that sales values have still not changed. 2. Switch to Model view. Edit (double-click) the relationship between **Salesperson** and **Sales** tables.      1. In the **Edit Relationship** window, uncheck the **Make This Relationship Active** checkbox and select **OK**.      1. In the diagram, notice that a dashed line represents an inactive relationship.   [Picture 5697](https://microsoftlearning.github.io/PL-300-Microsoft-Power-BI-Data-Analyst/Instructions/Labs/Linked_image_Files/04-configure-data-model-in-power-bi-desktop-advanced_image17.png)   1. Switch to Report view, and then notice that sales for Michael Blythe are now approximately $22 million.   Picture 5698   1. Switch to Model view and select the Salesperson table in the diagram. 2. In the Properties pane, replace text in the Name box with Salesperson (Performance). |

***Task 10: Relate Targets Table***

|  |
| --- |
| 1. Create a relationship from Salesperson (Performance) | EmployeeID column and Targets | EmployeeID column.      1. In the Report view, add the Targets | Target field to the table visual.      1. Resize the table visual so all columns are visible.   Picture 5699 |

**Lab 4-02: Create DAX Calculations in Power BI Desktop**

**Pre-requisites**

* Download and install Power BI from the Microsoft Store
* The lab links to a localhost SQL Server instance. Download a free Developer copy of the install media.

https://www.microsoft.com/sql-server/sql-server-downloads?SilentAuth=1&f=255&MSPPError=-2147217396&rtc=1

* Install SQL Server from Installation Wizard (Setup)

<https://learn.microsoft.com/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup>

* Install the latest version of Microsoft Edge to access Power BI service online
* Extract the ‘AllFiles’ folder to F:/ and rename it to ‘F:\Allfiles'

<https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/AllfilesDownload.zip>

**Introduction**

DAX, or Data Analysis Expressions, is a powerful formula language that lets you transform and analyze your data beyond what is readily available in tables. With DAX, you can create custom calculations, extend your data model's capabilities, and ultimately gain deeper insights from your data.

**Problem**

While your data might be organized, it might not always be presented in the format you need for analysis. For example, you might want to calculate sales growth year-over-year or analyze sales performance by specific regions. Basic table functionalities might not provide these insights directly.

**Solution**

Through this lab, you will learn how to leverage DAX functions, navigate table relationships, and write formulas to unlock the hidden potential of your data and create informative reports in Power BI.

Open the 04-Starter-Sales Analysis.pbix file.

***Task 1: Create the Salesperson Calculated Table***

|  |
| --- |
| 1. In Power BI Desktop, switch to Report view, and on the Modeling ribbon, select New Table from the Calculations group.   Picture 1   1. In the formula bar that appears below the ribbon when making or editing calculations, type Salesperson =, press Shift+Enter, then type 'Salesperson (Performance)', and press Enter.   Picture 4   1. In the Data pane, observe that the table icon now has an additional calculator symbol in front of it, indicating a calculated table.   Picture 10   1. Switch to Model view and verify that the Salesperson table is visible (you may need to reset the view to locate it). 2. Establish a relationship between the Salesperson | EmployeeKey column and the Sales | EmployeeKey column. 3. Right-click the inactive relationship between the Salesperson (Performance) and Sales tables and select Delete. |

***Task 2: Create the Date Table***

|  |
| --- |
| 1. Switch to Table view. On the Home ribbon tab, select New Table from the Calculations group.   Picture 5   1. In the formula bar, enter the following DAX:   Picture 6   1. Observe the column of date values, which are formatted according to US regional settings (i.e., mm/dd/yyyy).   Picture 7   1. At the bottom-left corner of the status bar, check the table statistics, which confirm that 1826 rows of data have been created, representing five full years’ worth of data. |

***Task 3: Create Calculated Columns***

|  |
| --- |
| 1. On the Table Tools contextual ribbon, select New Column from the Calculations group.   Picture 11   1. In the formula bar, type the following (or copy from the snippets file), and press **Enter**:      1. Use snippets file definitions to create the following two calculated columns for the **Date** table:  * Quarter * Month  1. Verify new columns are visible.   Picture 14   1. To validate calculations, switch to Report view. 2. To create a new report page, select the plus icon next to Page 1.   Picture 15   1. To add a matrix visual to the new report page, in the **Visualizations** pane, select the matrix visual type.   Picture 51   1. In the Data pane, within the Date table, drag the Year field into the Rows area.   Picture 17   1. Drag the Month field into the Rows area directly below the Year field. 2. In the top-right corner (or bottom, depending on the visual's position) of the matrix visual, click the forked double arrow icon to expand all years by one level.   Picture 19   1. Observe that the years now display their corresponding months, with the months sorted alphabetically rather than chronologically.   Picture 20   1. To adjust the sort order of the Month field, switch to Table view. Add the MonthKey column to the Date table.      1. In the Table view, confirm that the new column contains numeric values (e.g., 201707 for July 2017).   Picture 21   1. Return to Report view. In the Data pane, select the Month field. 2. On the Column Tools contextual ribbon, in the Sort group, choose Sort by Column and then select MonthKey.   Picture 22   1. In the matrix visual, notice that the months are now chronologically sorted.   Picture 23 |

***Task 4: Complete the Date Table***

|  |
| --- |
| 1. In the Data pane on the right side, select the Date table, right-click the Year column, and choose "Create Hierarchy." Rename the newly created hierarchy to "Fiscal" by right-clicking it and selecting "Rename."      1. Add the following two remaining fields to the Fiscal hierarchy by clicking them in the **Data** pane, right-clicking, and selecting **Add to hierarchy** -> **Fiscal**.  * Quarter * Month   [Picture 24](https://microsoftlearning.github.io/PL-300-Microsoft-Power-BI-Data-Analyst/Instructions/Labs/Linked_image_Files/05-create-dax-calculations-in-power-bi-desktop_image31.png)   1. Create the following two model relationships:  * Date | Date to Sales | OrderDate * Date | Date to Targets | TargetMonth      1. Hide the following two columns:  * Sales | OrderDate * Targets | TargetMonth |

***Task 5: Mark the Date Table***

|  |
| --- |
| 1. Switch to Report view. In the Data pane, select the Date table (not the Date field). On the Table Tools contextual ribbon, within the Calendars group, click "Mark as Date Table." 2. In the "Mark as Date Table" window, set the Mark as Date Table property to Yes. From the "Choose a date column" dropdown list, select Date, and then click Save.   Mark as date table   1. Save the Power BI Desktop file. |

***Task 6: Create Simple Measures***

|  |
| --- |
| 1. In Report view, on Page 2, drag the Sales | Unit Price field into the matrix visual from the Data pane.   Picture 27   1. In the visual fields pane (located below the Visualizations pane), observe that Unit Price is listed as the Average of Unit Price in the Values field well/area. Click the down-arrow next to Unit Price to view the available menu options.   Picture 30   1. To create a measure, right-click the Sales table in the Data pane and select "New Measure." 2. In the formula bar, enter the following measure definition:      1. In the **Values** well, open the context menu for the **Avg Price** field, and notice that it is not possible to change the aggregation technique.   Picture 32   1. Use the snippets file definitions to create the following five measures for the **Sales** table:  * Median Price      * Min Price      * Max Price      * Orders      * Order Lines      1. Switch to Model view and then select all four price measures. 2. For selected measures, configure the following settings:  * Set format to two decimal places * Assign to a display folder **Pricing**   Picture 33   1. Hide **Unit Price** column. 2. Multi-select **Order Lines** and **Orders** measures, and then configure the following requirements:  * Set the format using the thousands separator * Assign to a display folder named **Counts**   Picture 36   1. Add the following five measures to the matrix visual:  * Median Price * Min Price * Max Price * Orders * Order Lines  1. Verify that the table looks sensible and is correctly formatted.   Picture 39 |

***Task 7: Create Additional Measures***

|  |
| --- |
| 1. In Report view, click on Page 1 and review the table visual, paying attention to the total for the Target column.   Picture 41   1. Select the table visual, and in the Visualizations pane, remove the Target field. 2. Rename the Targets | Target column to Targets | TargetAmount.      1. Create the following measure on the **Targets** table:      1. Format **Target** measure for zero decimal places.      1. Hide the **TargetAmount** column.      1. Add **Target** measure to the table visual. 2. Notice that the **Target** column total is now BLANK.   Picture 43   1. Use the snippets file definitions to create the following two measures for the **Targets** table:  * Variance * Variance Margin  1. Format the Variance measure to display zero decimal places. 2. Format the Variance Margin measure as a percentage with two decimal places. 3. Add Variance and Variance Margin measures to the table visual. 4. Resize the table visual to ensure all columns and rows are visible.   Picture 44   1. In the top-right corner of the Data pane, collapse and then expand the pane. 2. Observe that the Targets table now appears at the top of the list.   Picture 46 |

**Lab 4-03: Create Advanced DAX Calculations in Power BI Desktop**

**Pre-requisites**

* Download and install Power BI from the Microsoft Store
* The lab links to a localhost SQL Server instance. Download a free Developer copy of the install media.

https://www.microsoft.com/sql-server/sql-server-downloads?SilentAuth=1&f=255&MSPPError=-2147217396&rtc=1

* Install SQL Server from Installation Wizard (Setup)

<https://learn.microsoft.com/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup>

* Install the latest version of Microsoft Edge to access Power BI service online
* Extract the ‘AllFiles’ folder to F:/ and rename it to ‘F:\Allfiles'

<https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/AllfilesDownload.zip>

**Introduction**

This lab dives into the world of advanced Data Analysis Expressions (DAX) for Power BI Desktop. DAX is a formula language that unlocks the power of your data by allowing you to create custom calculations and extend the capabilities of your data model. By mastering advanced DAX techniques, you can gain deeper insights and create more sophisticated analyses in Power BI.

**Problem**

Basic DAX expressions provide a solid foundation, but complex data analysis often requires more advanced techniques. This lab focuses on overcoming limitations of basic DAX by exploring functionalities like:

* Manipulating filter context for specific calculations
* Creating time intelligence measures for Year-over-Year (YoY) comparisons and Year-to-Date (YTD) calculations
* Utilizing advanced DAX functions for complex data transformations

**Solution**

This lab guides you through creating advanced DAX measures to tackle various scenarios. You will learn to leverage functions like CALCULATE, DATESYTD, and VAR to achieve your analytical goals. By working through these exercises, you will increase the skills to write robust DAX expressions that unlock the full potential of your Power BI data model.

Open the 05-Starter-Sales Analysis.pbix file. For your ease, all DAX definitions in this lab can be copied from the 05-advanced-dax\Snippets.txt file.

***Task 1: Create a Matrix Visual***

|  |
| --- |
| 1. In Power BI Desktop, **Report view**, generate new report. On **Page 3**, add a matrix visual.   Picture 13   1. Resize the matrix visual to fill the entire page. 2. To arrange matrix visual fields from the **Data** pane, drag the **Region | Regions** hierarchy and drop it inside the visual. Add also **Sales | Sales** field to Values well.      1. To expand the entire hierarchy, at the top-right of the matrix visual, select the forked-double arrow icon twice.   Picture 47   1. To format the visual, in the **Visualizations** pane, select the **Format** pane.   Picture 14   1. In the **Search** box, enter **Layout**. Set the **Layout** property to **Tabular**.   Picture 49   1. Verify that the matrix visual now has 4 column headers.   Picture 50 |

***Task 2: Manipulate Filter Context***

|  |
| --- |
| 1. Add a measure to the **Sales** table based on the following expression:      1. Add **Sales All Region** measure to matrix visual.   Picture 52   1. Observe that the Sales All Region measure calculates the total sales for each region, country (subtotal), and group (subtotal). 2. In the Data pane, make sure the Sales All Region measure is selected (it will have a dark gray background when selected). Then, in the formula bar, replace the measure name and formula with the following:      1. Format **Sales % All Region** measure as a percentage with two decimal places.      1. In the matrix visual, review **Sales % All Region** measure values.   Picture 53   1. Add a new measure to the Sales table using the following expression and format it as a percentage:      1. Add the Sales % Country measure to the matrix visual. Note that only the regions in the United States display a value different from 100%.   Picture 54   1. To improve the readability of this measure visual, overwrite the **Sales % Country** measure with this improved formula.      1. Observe that the Sales % Country measure now returns a value only when a region is in scope.   Picture 55   1. Add a new measure to the Sales table using the following expression and format it as a percentage:      1. Add the Sales % Group measure to the matrix visual. 2. To enhance the readability of this measure, replace the Sales % Group measure with the improved formula provided.      1. Observe that the Sales % Group measure now returns a value only when a region or country is in scope. 2. In Model view, organize the three new measures into a display folder named Ratios.      1. Save the Power BI Desktop file. |

***Task 3: Create a YTD Measure***

|  |
| --- |
| 1. In Report view, on Page 2, observe the matrix visual that shows various measures with years and months grouped in the rows. 2. Add a measure to the Sales table using the following expression and format it to zero decimal places:      1. Add the Sales field and the Sales YTD measure to the matrix visual. 2. Notice the accumulation of sales values throughout the year.   Picture 59 |

***Task 4: Create a YoY Growth Measure***

|  |
| --- |
| 1. Add another measure to the **Sales** table based on the following expression:      1. Add Sales YoY Growth measure to the matrix visual. 2. Observe that the new measure displays BLANK for the first 12 months (due to the absence of sales data before fiscal year 2017). Note that the Sales YoY Growth measure value for July 2018 matches the Sales value for July 2017.   Picture 61   1. To finalize the measure, replace the Sales YoY Growth measure with the following formula and format it as a percentage with two decimal places:      1. In the formula's RETURN clause, observe that the variable is referenced twice. 2. Confirm that the YoY growth for July 2018 is 392.83%.   Picture 62   1. In Model view, organize the two new measures into a display folder named Time Intelligence. |

**Lab 4-04: Enforce Model Security**

**Pre-requisites**

* Download and install Power BI from the Microsoft Store
* The lab links to a localhost SQL Server instance. Download a free Developer copy of the install media.

https://www.microsoft.com/sql-server/sql-server-downloads?SilentAuth=1&f=255&MSPPError=-2147217396&rtc=1

* Install SQL Server from Installation Wizard (Setup)

<https://learn.microsoft.com/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup>

* Install the latest version of Microsoft Edge to access Power BI service online
* Extract the ‘AllFiles’ folder to F:/ and rename it to ‘F:\Allfiles'

<https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/AllfilesDownload.zip>

**Introduction**

Sharing data is crucial for collaboration, but ensuring the right people see the right information is equally important. This lab delves into the world of Power BI model security, equipping you with the skills to control data access within your reports.

**Problem**

In collaborative environments, it is essential to restrict access to sensitive data. For instance, a sales report might contain revenue figures that only specific teams should be able to view. Sharing the entire report with everyone poses a security risk.

**Solution**

This lab provides a hands-on approach to setting up user roles, defining security rules, and configuring your data model for a secure and collaborative BI experience.

This lab has been done by using a VM, which was already created in the below link: https://learn.microsoft.com/en-us/training/modules/enforce-power-bi-model-security/4a-exercise-enforce-model-security

|  |
| --- |
| 1. On the start menu, open Command Prompt.      1. In the command prompt window, navigate to D drive by typing:      1. In the command prompt window, enter the following command to download the course files and save them to a folder called DP500.   git clone [https://github.com/MicrosoftLearning/DP-500-Azure-Data-Analyst DP500](https://github.com/MicrosoftLearning/DP-500-Azure-Data-Analyst%20DP500)     1. To open File Explorer, on the taskbar, select the **File Explorer** shortcut. Go to the **D:\DP500\Allfiles\09\Starter** folder. 2. To open a pre-developed Power BI Desktop file, double-click the **Sales Analysis - Enforce model security.pbix** file.      1. If you are not already signed in, at the top-right corner of Power BI Desktop, select **Sign In**. Use the lab credentials to complete the sign-in process.      1. To save the file, on the **File** ribbon, select **Save as**. 2. In the **Save As** window, go to the **D:\DP500\Allfiles\09\MySolution** folder. Select **Save**. 3. In Power BI Desktop, at the left, switch to **Model** view.      1. Use the model diagram to review the model design.      1. Expand open the **Sales Territory** table.      1. Notice that the table includes a **Region** column.      1. Switch to **Report** view.      1. In the stacked column chart visual, in the legend, notice (for now) that it is possible to see many regions.      1. To add a security role, on the **Modeling** ribbon tab, from inside the **Security** group, select **Manage roles**.      1. In the **Manage roles** window, select **Create**.      1. To name the role, replace the selected text with **Australia**, and then press **Enter**.      1. In the **Tables** list, for the **Sales Territory** table, select the ellipsis, and then select **Add filter** > **[Region]**.      1. In the Table filter DAX expression box, replace Value with Australia.      1. Press **Create** to create another role.      1. Repeat the steps in this task to create a role named **Canada** that filters the **Region** column by **Canada**.      1. Select **Save**.      1. On the **Modeling** ribbon tab, from inside the **Security** group, select **View as**.      1. In the **View as roles** window, select the **Australia** role.      1. Select **OK**.      1. On the report page, notice that the stacked column chart visual shows only data for Australia.      1. Across the top of the report, notice the yellow banner that confirms the enforced role.      1. To stop viewing by using the role, at the right of the yellow banner, select **Stop viewing**.      1. Save the Power BI Desktop file.      1. To publish the report, on the **Home** ribbon tab, select **Publish**.      1. In the **Publish to Power BI** window, select your workspace, and then select **Select**.      1. When the publishing succeeds, select **Got it**.      1. Switch to the Power BI service (web browser). 2. In the workspace landing page, notice the **Sales Analysis - Enforce model security** dataset.      1. Hover the cursor over the dataset, and when the ellipsis appears, select the ellipsis and then select **Security**.      1. At the left, notice the list of roles, and that **Australia** is selected.      1. In the **Members** box, commence entering **Salespeople\_Australia**.      1. Select **Add**.      1. To complete the role mapping, select **Save**.      1. To return to the workspace landing page, in the **Navigation** pane, select the workspace. 2. To remove the dataset, hover the cursor over the dataset, and when the ellipsis appears, select the ellipsis and then select **Delete**.      1. When prompted to confirm deletion, select **Delete**.      1. Switch to Power BI Desktop. 2. To remove the security roles, on the **Modeling** ribbon tab, from inside the **Security** group, select **Manage roles**.      1. In the **Manage roles** window, to remove the first role, select **Delete**.      1. When prompted to confirm the deletion, press **Yes, delete**.      1. Also, remove the second role. 2. Select **Save**.      1. Switch to **Model** view.      1. On the **Home** ribbon tab, from inside the **Queries** group, select the **Transform data** icon.      1. In the **Power Query Editor** window, in the **Queries** pane (located at the left), right-click the **Customer** query, and then select **Duplicate**.      1. In the **Query Settings** pane (located at the right), in the **Name** box, replace the text with **Salesperson**.      1. In the **Applied Steps** list, right-click the **Removed Other Columns** step (third step), and then select **Delete Until End**.      1. When prompted to confirm deletion of the step, select **Delete**.      1. To source data from a different data warehouse table, in the **Applied Steps** list, in the **Navigation** step (second step), select the gear icon (located at the right).      1. In the **Navigation** window, select the **DimEmployee** table.      1. Select **OK**.      1. To remove unnecessary columns, on the **Home** ribbon tab, from inside the **Manage Columns** group, select the **Choose Columns** icon.      1. In the **Choose Columns** window, uncheck the **(Select All Columns)** item.      1. Check the following three columns:  * EmployeeKey * SalesTerritoryKey * EmailAddress  1. Select **OK**.      1. To rename the **EmailAddress** column, double-click the **EmailAddress** column header. 2. Replace the text with **UPN**, and then press **Enter**.      1. To load the table to the model, on the **Home** ribbon tab, select the **Close & Apply** icon.      1. When the table has been added to the model, notice that a relationship to the **Sales Territory** table was automatically created. 2. Right-click the relationship between the **Salesperson** and **Sales Territory** tables, and then select **Properties**.      1. In the **Edit relationship** window, in the **Cross filter direction** dropdown list, select **Both**. 2. Check the **Apply security filter in both directions** checkbox.      1. Select **OK**.      1. To hide the table, at the top-right of the **Salesperson** table, select the eye icon.      1. Switch to **Report** view.      1. To add a security role, on the **Modeling** ribbon tab, from inside the **Security** group, select **Manage roles**.      1. In the **Manage roles** window, select **Create**.      1. To name the role, replace the selected text with **Salespeople**.      1. This time, only one role needs to be created. 2. Add a filter to the **UPN** column of the **Salesperson** table.      1. In the Table filter DAX expression box, replace "Value" with USERPRINCIPALNAME().      1. Select **Save**.      1. On the **Modeling** ribbon tab, from inside the **Security** group, select **View as**.      1. In the **View as roles** window, check **Other user**, and then in the corresponding box, enter **michael9@adventure-works.com**      1. Check the **Salespeople** role.      1. Select **OK**.      1. On the report page, notice that the stacked column chart visual shows only data for the Northeast.      1. Across the top of the report, notice the yellow banner that confirms the enforced role.      1. To stop viewing by using the role, at the right of the yellow banner, select **Stop viewing**.      1. Save the Power BI Desktop file.      1. Publish the report to the workspace you created at the beginning of the lab. 2. Close Power BI Desktop. 3. Switch to the Power BI service (web browser). 4. Go to the security settings for the **Sales Analysis - Enforce model security** dataset. 5. Map the **Salespeople** security group the **Salespeople** role. |