## Lab 3-01: Get Data 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
* All files can be downloaded from GitHub. 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 introduces you to the fundamentals of working with data in Power BI Desktop. You will explore how to connect to various data sources, preview the data to understand its structure and quality, and leverage data profiling features for deeper analysis.

### Problem

Many business analysts struggle with efficiently acquiring and understanding data before diving into creating reports and visualizations. Power BI Desktop provides a robust set of tools to connect to diverse data sources, but users need to be familiar with the process of data retrieval and exploration.

### Solution

By the end of this lab, you will be comfortable retrieving and understanding data from various sources, setting the stage for data transformation and visualization in Power BI Desktop.

### Solution

|  |
| --- |
| **Get started with Power BI Desktop**  To complete this lab, first open a web browser and enter the following URL to download the zip folder:  [**https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/Allfiles/Labs/01-prepare-data-with-power-query-in-power-bi-desktop/01-prepare-data.zip**](https://github.com/MicrosoftLearning/PL-300-Microsoft-Power-BI-Data-Analyst/raw/Main/Allfiles/Labs/01-prepare-data-with-power-query-in-power-bi-desktop/01-prepare-data.zip)    Extract the folder to the **C:\Users\Hania Ahmed\Downloads\01-prepare-data** folder.    Open the **01-Starter-Sales Analysis.pbix** file.     * This starter file has been specially configured to help you complete the lab. The following report-level settings have been disabled in the starter file:   + Data Load > Import relationships from data sources on the first load   + Data Load > Autodetect new relationships after data is loaded   **Get data from SQL Server**  This task teaches you how to connect to an SQL Server database and import tables, which create queries in Power Query.   1. Select **SQL Server** from inside the **Data** group on the **Home** ribbon tab.      1. In the **SQL Server Database** window, in the **Server** box, enter **localhost** and leave **Database** blank, then select **OK**.     **Note**: In this lab, you will connect to the SQL Server database using **localhost** because gateway data sources cannot resolve **localhost**. This is not a recommended practice when creating your own solutions.   1. Select **Windows** > **Use my** **current credentials** if prompted for credentials, then **Connect**.      1. Select **OK** if you receive a warning that an encrypted connection cannot be established.      1. In the **Navigator** pane, expand the **AdventureWorksDW2020** database.     **Note**: The **AdventureWorksDW2020** database is based on the **AdventureWorksDW2017** sample database. It has been modified to support the learning objectives of the course labs.   1. Select the **DimEmployee** table, and notice the preview of the table data.     **Note**: The preview data lets you see the columns and a sample of rows.   1. To import the table data, **select the checkbox** next to the following tables:    * DimEmployee    * DimEmployeeSalesTerritory    * DimProduct    * DimReseller    * DimSalesTerritory    * FactResellerSales     Complete this task by selecting **Transform Data**, which will open Power Query Editor - leave this open for the next task.    You have now connected to six tables from an SQL Server database.    **Preview Data in Power Query Editor**  This task introduces the Power Query Editor and allows you to review and profile the data. This helps you determine how to clean and transform the data later. You will also review both dimension tables prefixed with "Dim" and fact tables prefixed with "Fact".   1. At the left, notice the **Queries** pane in the **Power Query Editor** window. The **Queries** pane contains one query for each table you checked.      1. Select the first query—**DimEmployee**.     The **DimEmployee** table in the SQL Server database stores one row for each employee. A subset of the rows from this table represents the salespeople, which will be relevant to the model you will develop.   1. At the bottom left corner of the status bar, some table statistics are provided—the table has 33 columns and 296 rows.      1. In the data preview pane, scroll horizontally to review all columns. Notice that the last five columns contain **Table** or **Value** links.     These five columns represent relationships to other tables in the database. They can be used to join tables together. You will join tables in the **Load Transformed Data in Power BI Desktop** lab.   1. Check **Column Quality** on the **View** ribbon tab inside the **Data Preview** group to assess column quality. The column quality feature lets you easily determine the percentage of valid, error, or empty values in columns.      1. Notice that the **Position** column has 94% empty (null) rows.      1. Check **Column Distribution** on the **View** ribbon tab inside the **Data Preview** group to assess column distribution.      1. Review the **Position** column again and notice four distinct values and one unique value.      1. Review the column distribution for the **EmployeeKey** column—there are 296 distinct and 296 unique values.     **Note**: When the distinct and unique counts are the same, the column contains unique values. When modeling, some model tables must have unique columns. These unique columns can create one-to-many relationships, which you will do in the **Model Data in Power BI Desktop** lab.   1. In the **Queries** pane, select the **DimProduct** query.     The **DimProduct** table contains one row per product sold by the company.     1. In the **Queries** pane, select the **DimReseller** query.     The **DimReseller** table contains one row per reseller. Resellers sell, distribute, or value add to the Adventure Works products.     1. Check **Column Profile** on the **View** ribbon tab inside the **Data Preview** group to view column values.      1. Select the **BusinessType** column header, and notice the new pane beneath the data preview pane. Review the column statistics and value distribution in the data preview pane.     Notice the data quality issue: two labels for the warehouse (**Warehouse** and the misspelled **Ware House**).   1. Hover the cursor over the **Ware House** bar and notice five rows with this value.      1. In the **Queries** pane, select the **DimSalesTerritory** query.     The **DimSalesTerritory** table contains one row per sales region, including **Corporate HQ** (headquarters). Regions are assigned to a country, and countries are assigned to groups. You will create a hierarchy in the **Model Data in the Power BI Desktop** lab to support regional, country, or group analysis analysis.   1. In the **Queries** pane, select the **FactResellerSales** query.     The **FactResellerSales** table contains one row per sales order line—a sales order contains one or more line items.   1. Review the column quality for the **TotalProductCost** column and notice that 8% of the rows are empty.     Missing **TotalProductCost** column values is a data quality issue.  **Get data from a CSV file**  You will create a new query based on CSV files in this task.   1. To add a new query, in the **Power Query Editor** window, on the **Home** ribbon tab, select the **New Source** down-arrow from inside the **New Query** group and then **Text/CSV**.      1. Navigate to the **01-prepare-data** > **ResellerSalesTargets.csv** file. Select **Open**.      1. In the **ResellerSalesTargets.csv** window, review the preview data. Select **OK**.      1. In the **Queries** pane, notice the addition of the **ResellerSalesTargets** query.     The **ResellerSalesTargets** CSV file contains one row per salesperson per year. Each row records 12 monthly sales targets (expressed in thousands). The business year for the Adventure Works company commences on July 1.   1. Notice that no column contains empty values. When there is no monthly sales target, a hyphen character is stored instead.      1. Review the icons in each column header to the left of the column name. The icons represent the column data type. **123** is the whole number, and **ABC** is the text.      1. Repeat the steps to create a query based on the **ColorFormats.csv** file.     The **ColorFormats** CSV file contains one row per product color. Each row records the HEX codes to format the background and font colors.  You should now have two new queries, **ResellerSalesTargets** and **ColorFormats**. |

## Lab 3-02: Load Transformed Data 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
* All files can be downloaded from GitHub. 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 the previous lab, you conquered the initial hurdle of acquiring data in Power BI Desktop. Now, this lab delves into the world of data transformation, a crucial step in preparing your data for insightful analysis.

### Problem

Raw data rarely aligns perfectly with your analytical needs. It might contain inconsistencies, irrelevant columns, or require restructuring. Manually manipulating data outside of Power BI can be cumbersome and error-prone.

### Solution

Through this lab, you will master the art of data transformation in Power BI Desktop, preparing your data to become the foundation for compelling and informative visualizations.

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

#### Task 1: Configure the Salesperson Query

|  |
| --- |
| 1. To open the Power Query Editor window, 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, select the DimEmployee query.      1. To rename the query, in the Query Settings pane (located at the right), in the Name box, replace the text with Salesperson, and then press Enter. Then verify the name has been updated in the Queries pane.      1. To find a specific column, on the Home ribbon tab, select the Manage Columns down-arrow, select Choose Columns down-arrow, and then click Go to Column.      1. In the Go to Column window, to order the list by column name, select the AZ sort button, and then select Name and SalesPersonFlag. Click OK.      1. Locate the SalesPersonFlag column, then filter the column to select only Salespeople (that is, TRUE), and click OK.      1. In the Query Settings pane, in the Applied Steps list, notice the addition of the Filtered Rows step.      1. To remove columns, on the Home ribbon tab, select the Manage Columns group and select the Choose Columns icon.      1. To include columns, check the following six columns:      1. In the Applied Steps list, notice the addition of another query step.      1. To create a single name column, first select the FirstName column header. While pressing the Ctrl key, select the LastName column.      1. Right-click either of the selected column headers, and then, in the context menu, select Merge Columns.      1. In the Merge Columns window, in the Separator dropdown list, select Space. In the New Column Name box, replace text with Salesperson.      1. To rename the EmployeeNationalIDAlternateKey column, double-click the EmployeeNationalIDAlternateKey column header, replace text with EmployeeID, and then press Enter.      1. Use previous steps to rename the EmailAddress column to UPN.      1. At the bottom-left, in the status bar, verify that the query has 5 columns and 18 rows. |

#### Task 2: Configure the SalespersonRegion Query

|  |
| --- |
| 1. In the **Queries** pane, select the **DimEmployeeSalesTerritory** query. In the **Query Settings** pane, rename the query to **SalespersonRegion**.      1. To remove the last two columns, first select the **DimEmployee** column header. While pressing the **Ctrl** key, select the **DimSalesTerritory** column header.      1. Right-click either of the selected column headers and then in the context menu, select **Remove Columns**. |

#### Task 3: Configure the Product Query

|  |
| --- |
| 1. Select the **DimProduct** query and rename the query to **Product**.      1. Locate the **FinishedGoodsFlag** column, and then filter the column to retrieve products that are finished goods (that is, TRUE).      1. Remove all columns **except** the following:        1. In the **DimProductSubcategory** column header, at the right of the column name, select expand button.   Column expand icon   1. See the full list of columns, then select the **Select All Columns** box to unselect all columns.      1. Select **EnglishProductSubcategoryName** and **DimProductCategory**, and uncheck the **Use Original Column Name as Prefix** checkbox before selecting **OK**.      1. Notice that the transformation resulted in the addition of two columns and that the **DimProductSubcategory** column has been removed.      1. Expand the **DimProductCategory** column, and then introduce only the **EnglishProductCategoryName** column.      1. Rename the following four columns: |

#### Task 4: Configure the Reseller Query

|  |
| --- |
| 1. Select the **DimReseller** query and rename it to **Reseller**.      1. Remove all columns **except** the following:      1. Expand the **DimGeography** column to include **only** the following three columns:      1. On the **Business Type** column header, select the down-arrow, and then review separate column values, and notice both values **Warehouse** and **Ware House**.      1. Right-click the **Business Type** column header, and then select **Replace Values**.      1. In the **Replace Values** window, configure the following values:      1. Rename the following columns: |

#### Task 5: Configure the Region Query

|  |
| --- |
| 1. Select the **DimSalesTerritory** query and rename the query to **Region**.      1. Apply a filter to the **SalesTerritoryAlternateKey** column to remove the value 0 (zero).      1. Remove all columns **except** the following:      1. Rename the following three columns: |

#### Task 6: Configure the Sales Query

|  |
| --- |
| 1. Select the **FactResellerSales** query and rename it to **Sales**.      1. Remove all columns **except** the following:      1. Expand the **DimProduct** column, uncheck all columns, and then include only the **StandardCost** column.      1. To create a custom column, on the **Add Column** ribbon tab from inside the **General** group, select **Custom Column**.   Picture 5664   1. In the **Custom Column** window, in the **New Column Name** box, replace the text with **Cost**. In the **Custom Column Formula** box, enter the following expression (after the equals symbol), then save the new column:      1. Remove the following two columns:      1. Rename the following three columns:  * OrderQuantity to Quantity * UnitPrice to Unit Price (include a space) * SalesAmount to Sales  1. To modify column data type, in the **Quantity** column header, at the left of the column name, select the **1.2** icon and then choose **Whole Number**.   Picture 5667   1. Modify the following three column data types to **Fixed Decimal Number**. |

#### Task 7: Configure the Targets Query

|  |
| --- |
| 1. Select the **ResellerSalesTargets** query and rename it to **Targets**.      1. To unpivot the 13-month columns (**M03-M12**), first, multi-select the **Year** and **EmployeeID** column headers.      1. Right-click either of the selected column headers and then in the context menu, select **Unpivot Other Columns**.      1. Notice that the column names now appear in the **Attribute** column, and the values appear in the **Value** column.      1. Apply a filter to the **Value** column to remove hyphen (-) values.      1. Rename the following two columns:      1. To prepare **MonthNumber** column values, right-click the **MonthNumber** column header and then select **Replace Values**.      1. In the **Replace Values** window, in the **Value To Find** box, enter **M** and leave **Replace with** empty.      1. Modify the **MonthNumber** column data type to **Whole Number**.      1. On the **Add Column** ribbon tab, from inside the **General** group, select The **Column From Examples** icon.   Picture 5675   1. Notice that the first row is for the year **2017** and month number **7**.      1. In the **Column1** column, in the first grid cell, commence entering **7/1/2017**, and then press **Enter**. Notice that the grid cells update with predicted values.      1. Notice the formula presented above the query grid.   Picture 5679   1. To rename a new column, double-click the **Merged** column header and rename the column as **TargetMonth**.      1. Remove the following columns:      1. Modify the following column data types:  * Target as a fixed decimal number * TargetMonth as date      1. To multiply the **Target** values by 1000, select the **Target** column header, and then on the **Transform** ribbon tab, from inside the **Number Column** group, select **Standard**, and then select **Multiply**.   Picture 5682   1. In the **Multiply** window, in the **Value** box, enter **1000**, and select **OK**. |

#### Task 8: Configure the ColorFormats Query

|  |
| --- |
| 1. Select the **ColorFormats** query and notice that the first row contains column names.      1. On the **Home** ribbon tab, from inside the **Transform** group, select **Use First Row as Headers**.   Picture 5688 |

#### Task 9: Update the Product Query

|  |
| --- |
| 1. Select **Product** query.      1. To merge the **ColorFormats** query, on the **Home** ribbon tab, select the **Combine** down arrow, then select **Merge Queries**.   Picture 5654   1. In the **Merge** window, in the **Product** query grid, select the **Color** column header.   Picture 5655   1. Beneath the **Product** query grid, in the dropdown list, select **ColorFormats** query.   Picture 21   1. In the **ColorFormats** query grid, select the **Color** column header.      1. When the **Privacy Levels** window opens, for each of the two data sources, in the corresponding dropdown list, select **Organizational**, then **Save**.   Picture 5691   1. In the **Merge** window, use default **Join Kind** - maintaining a selection of Left Outer and select **OK**.      1. Expand the **ColorFormats** column to include the following two columns: |

#### Task 10: Update the ColorFormats Query

|  |
| --- |
| 1. Select **ColorFormats** query.      1. In the **Query Settings** pane, select the **All Properties** link.   Picture 322   1. In the **Query Properties** window, uncheck the **Enable Load To Report** checkbox.   Picture 323   1. In Power Query Editor, confirm that you have **8 queries**:      1. Select **Close & Apply** to load data.   Picture 326   1. You can now see the canvas in Power BI Desktop, with Filters, Visualizations, and Data panes on the right. In the Data pane, notice the **7 tables** loaded to the data model.   Picture 3 |