Exercise: Configure data sources

Introduction

In this exercise, you'll use Power BI to help Tailwind Traders navigate through its data, apply transformations, and prepare its data for future analysis.

By completing this exercise, you'll demonstrate your ability to:

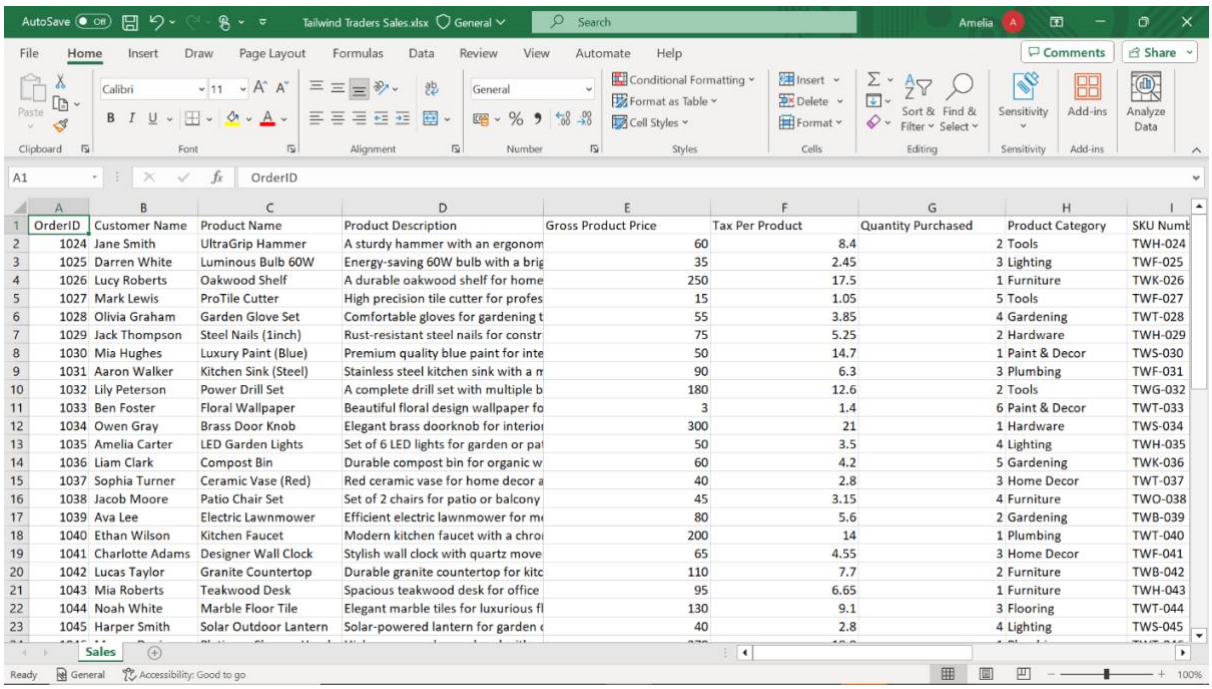
- Incorporate Excel data sources into Power BI and ensure accurate data types.
- Review statistics and details of the data sources to detect and rectify any discrepancies or errors.
- Identify and filter out refunded purchases to maintain the data's integrity.
- Implement a Python script to transform and refine the currency exchange data.

Case study

Tailwind Traders needs to configure its data sources. The company has sought your help and needs you to add sales data, ensure the accuracy of its data types, and transform its historical currency exchange data.

Instructions

Open the Excel workbook **Tailwind Traders Sales.xlsx** you created in the previous exercise. The workbook should contain a single **Sales** worksheet.



OrderID	Customer Name	Product Name	Product Description	Gross Product Price	Tax Per Product	Quantity Purchased	Product Category	SKU Numb
1024	Jane Smith	UltraGrip Hammer	A sturdy hammer with an ergonom	60	8.4	2	Tools	TWH-024
1025	Darren White	Luminous Bulb 60W	Energy-saving 60W bulb with a brig	35	2.45	3	Lighting	TWF-025
1026	Lucy Roberts	Oakwood Shelf	A durable oakwood shelf for home	250	17.5	1	Furniture	TWK-026
1027	Mark Lewis	ProTile Cutter	High precision tile cutter for profes	15	1.05	5	Tools	TWF-027
1028	Olivia Graham	Garden Glove Set	Comfortable gloves for gardening t	55	3.85	4	Gardening	TWT-028
1029	Jack Thompson	Steel Nails (1inch)	Rust-resistant steel nails for constr	75	5.25	2	Hardware	TWH-029
1030	Mia Hughes	Luxury Paint (Blue)	Premium quality blue paint for inte	50	14.7	1	Paint & Decor	TWS-030
1031	Aaron Walker	Kitchen Sink (Steel)	Stainless steel kitchen sink with a n	90	6.3	3	Plumbing	TWF-031
1032	Lily Peterson	Power Drill Set	A complete drill set with multiple b	180	12.6	2	Tools	TWG-032
1033	Ben Foster	Floral Wallpaper	Beautiful floral design wallpaper fo	3	1.4	6	Paint & Decor	TWT-033
1034	Owen Gray	Brass Door Knob	Elegant brass doorknob for interior	300	21	1	Hardware	TWS-034
1035	Amelia Carter	LED Garden Lights	Set of 6 LED lights for garden or pat	50	3.5	4	Lighting	TWH-035
1036	Liam Clark	Compost Bin	Durable compost bin for organic w	60	4.2	5	Gardening	TWK-036
1037	Sophia Turner	Ceramic Vase (Red)	Red ceramic vase for home decor a	40	2.8	3	Home Decor	TWT-037
1038	Jacob Moore	Patio Chair Set	Set of 2 chairs for patio or balcony	45	3.15	4	Furniture	TWO-038
1039	Ava Lee	Electric Lawnmower	Efficient electric lawnmower for m	80	5.6	2	Gardening	TWB-039
1040	Ethan Wilson	Kitchen Faucet	Modern kitchen faucet with a chro	200	14	1	Plumbing	TWT-040
1041	Charlotte Adams	Designer Wall Clock	Stylish wall clock with quartz move	65	4.55	3	Home Decor	TWF-041
1042	Lucas Taylor	Granite Countertop	Durable granite countertop for kitc	110	7.7	2	Furniture	TWB-042
1043	Mia Roberts	Teakwood Desk	Spacious teakwood desk for office	95	6.65	1	Furniture	TWH-043
1044	Noah White	Marble Floor Tile	Elegant marble tiles for luxurious fl	130	9.1	3	Flooring	TWT-044
1045	Harper Smith	Solar Outdoor Lantern	Solar-powered lantern for garden	40	2.8	4	Lighting	TWS-045

Step 1: Load the Sales data

1. Load the **Tailwind Traders Sales** file into Power BI and select **Transform**.

2. Within Power Query, find the **OrderID** column and set the data type to **Whole Number**.

3. To complete optimization, assign the following data types for the columns:

- **Gross Product Price = Fixed Decimal Number**
- **Tax Per Product = Fixed Decimal Number**
- **Quantity Purchased = Whole Number**
- **Loyalty Points = Whole Number**
- **Stock = Whole Number**
- **Product Category = Text**
- **Rating = Fixed Decimal Number**

4. In the **View** tab, upon selecting the **Column Quality**, **Column Distribution**, and **Column Profile**boxes, ensure the **Valid** percentage is **100%** for the **OrderID** column.

5. Select the **Gross Product Price** column and note down the histogram frequency of distinct and unique values.

6. Select the **Quantity Purchased** column and note down the **MIN**, **MAX** and **AVERAGE** values displayed on the additional statistical pane.

Step 2: Load the Purchases data

1. Load the **Purchases** file into Power BI and select **Transform**.
2. To complete optimization, assign the following data types for the columns: • **PurchaseID = Whole Number • OrderID = Whole Number • Return Policy (Days) = Whole Number • Purchase Date = Date • Warranty (Months) = Whole Number • Supplier = Text • Last Visited = Date • ReturnStatus = Text**

3. Select the **Warranty (Months)** column and note down the **MIN**, **MAX** and **AVERAGE** values displayed on the additional statistical pane.

4. Select the **ReturnStatus** column and observe the **Column Quality** pane to ensure the **Valid** percentage is **100%**.

5. Filter the **ReturnStatus** column to ensure that only records with **Not Returned** are visible.

Step 3: Load the Countries data

1. Load the **Countries** file into Power BI and select **Transform**.

2. To complete optimization, assign the following data types for the columns:

- **Country ID = Whole Number**
- **Exchange ID = Whole Number**
- **Country = Text**

Step 4: Load the Historical currency exchange data

1. Select **Get Data**, choose **Python** script, and then paste the following code into the script window in Power BI:

```
import pandas as pd
from io import StringIO

data = """Exchange ID;ExchangeRate;Exchange Currency
1;1;USD
2;0,75;GBP
3;0,85;EUR
4;3,67;AED
5;1,3;AUD"""
df = pd.read_csv(StringIO(data), sep=';')

# Return the transformed dataframe
df
```

2. Integrate this data into your Power BI report.

3. Save the Power BI project as **Tailwind Traders Report.pbix**.

Note: The Python script prepares the currency exchange data for analysis. It transforms the raw string data into a structured format that can be easily integrated with other datasets within Power BI. The core script elements are as follows:

- The **pandas** data analysis library is used for manipulating and analyzing data.
- **StringIO** is a module that lets you read and write strings like files.
- **pd.read_csv()** is a pandas function that reads a CSV file into a DataFrame.

Conclusion

Having completed the assigned tasks, you now have the skills to prepare data in an Excel report. You've also demonstrated the ability to perform calculations in your worksheet.