Microsoft Fabric

Fabric Analyst in a Day
Lab 7

Version: August 2024

Contents

Introduction

We are introduced to Lakehouse, ingested data from different data sources into the Lakehouse, set a refresh schedule for the data sources and created a data model. Now we are going to create a report.

By the end of this lab, you will have learned:

- How to auto-create a report
- How to build a report starting from a blank canvas
- How to build a report using Power BI Desktop
- How to experience Direct Lake mode resulting in data automatically refreshing

Power BI

Task 1: Auto-Create Report
Let's start by using the auto-create report option. And later in the lab, we will re-create the report we have in Power BI.

1. Let's navigate back to the **Fabric workspace** you created in the earlier lab.
2. From the bottom of the left panel select **Fabric experience selector** icon.
3. Fabric experience dialog opens. Select **Power BI**. You will be navigated to **Power BI Home page**.
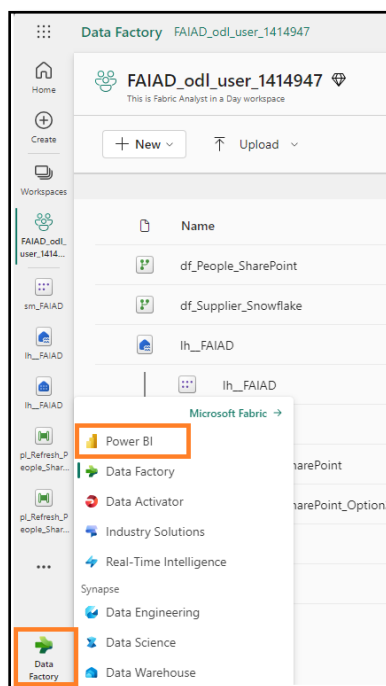

4. Select **New Report** from the top menu.
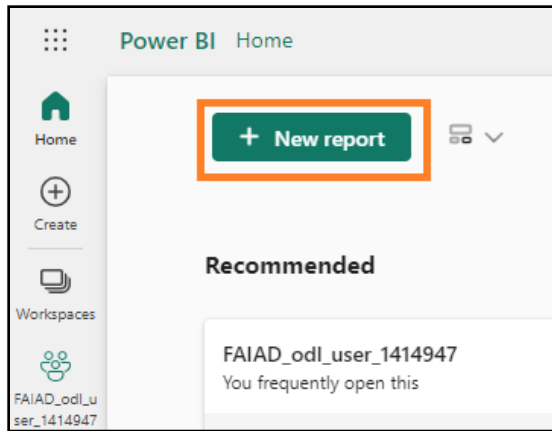
5. You will be navigated to **Build your first report screen**. There will be options to build report using excel, csv, enter data manually or to pick a published semantic model. We have created a semantic model in the previous labs. Let's use that. Select **Pick a published semantic model** option.
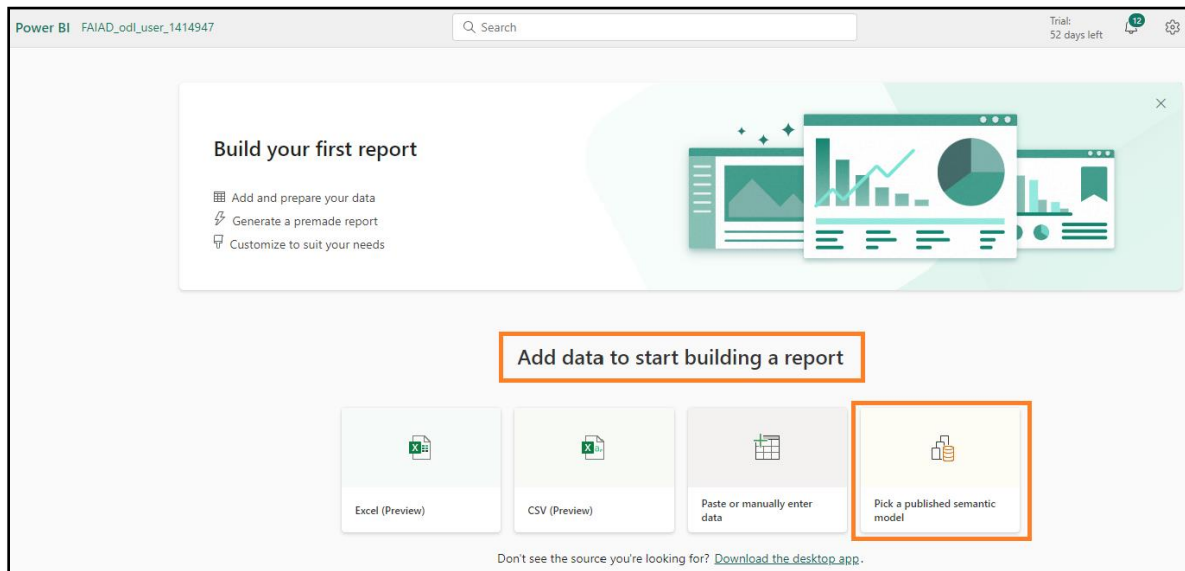
6. Pick a dataset to use in your report page opens. Notice we have four options. **Select sm_FAIAD**.
   a. **sm_FAIAD:** This is the semantic model we have created and want to use to build the report.
   b. **lh_FAIAD:** This is the lakehouse where we ingested all the data into.
   c. **Units by Supplier:** This is the dataset we created using T-SQL.
   d. **DataflowsStagingWarehouse:** This is the staging warehouse which is created by default. We did not use this as we did not stage data.
   e. **DataflowsStagingLakehouse:** This is the staging lakehouse which is created by default. We did not use this as we did not stage data.
7. Click the **arrow next to Auto-create report button**. Notice there are two options, Auto-create report and Create a blank report. Let's try auto-creating, so select **Auto-create report**.
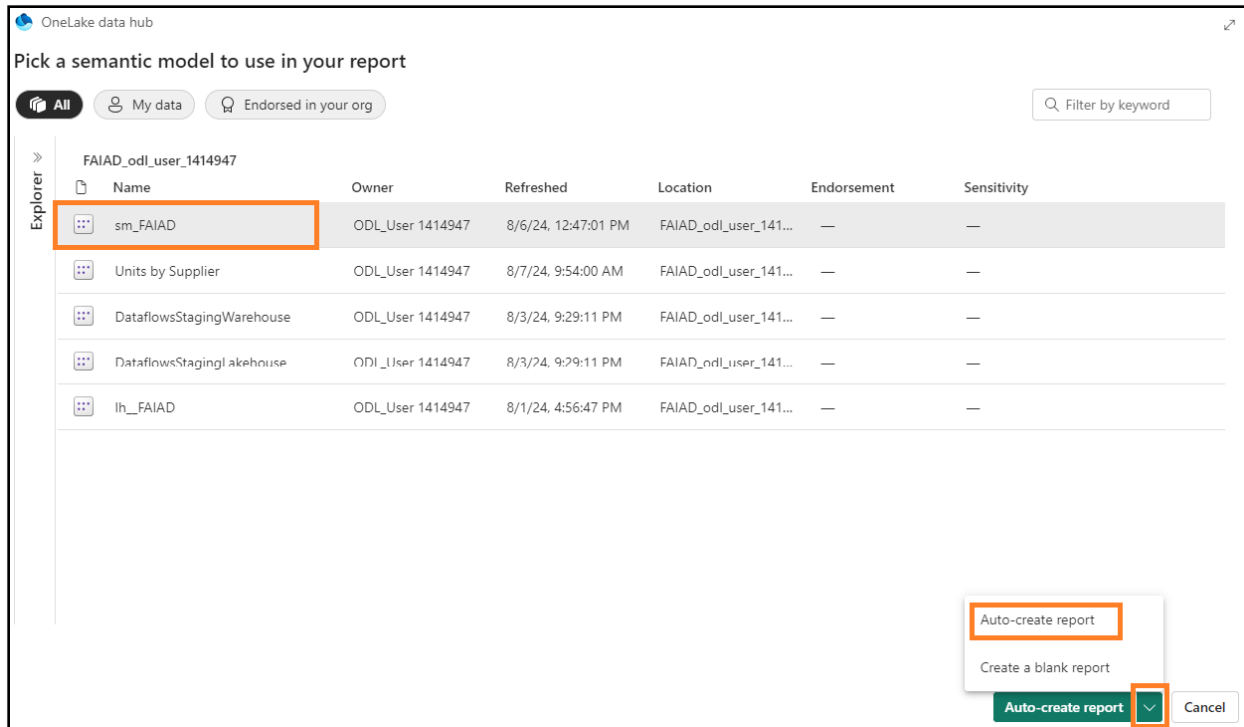
8. Power BI will start auto creating the report. Notice there is an option to Pre-select data, if we choose too. Once the report is ready, a dialog appears on the top right of the screen. Select **View report now or it will autoload in a few seconds**.
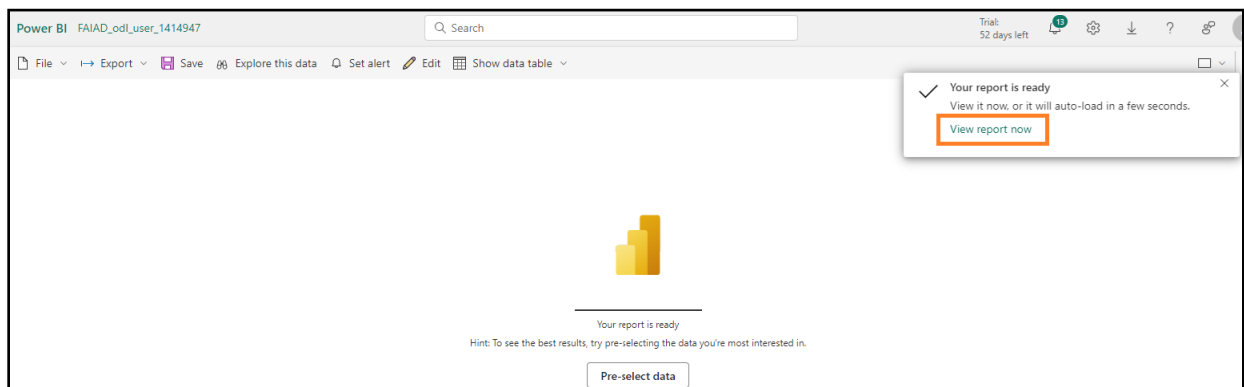
Checkpoint: You will have a report which looks like the screenshot below. There are a few KPIs and some trend visuals. This is a good start if you are analyzing a new model and need a jumpstart.

Note: Notice on the top menu, you have the option to Edit the report or view some of the data as tables. Feel free to explore these options.
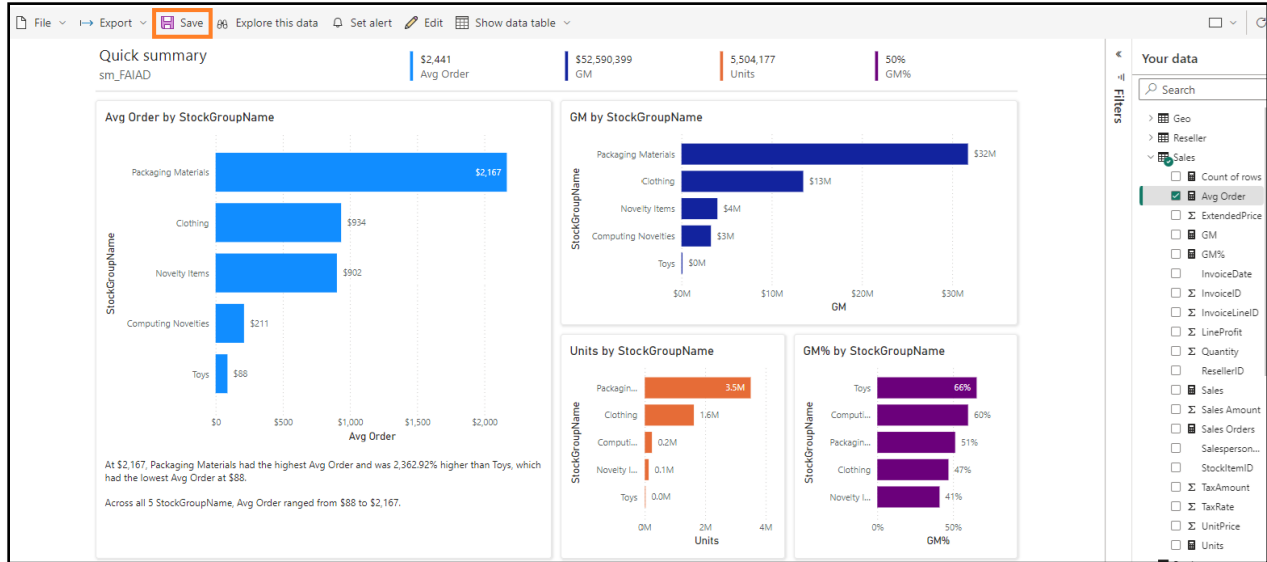
9. Let's save this report. From the top menu, select **Save**.
10. Save your report dialog opens. Name the report as **rpt_Sales_Auto_Report**
    Note: we are prefixing report name with rpt which is short for report.
11. Make sure the report is saved in your workspace, **FAIAD_<username>.**

12. Select **Save.**

Note: Auto-created report may look different for you as it is "auto-created". It also depends on the relationships and measures you created in the previous lab (Lab 6).

Above screenshot is how the auto-created report **may** look if you created all the relationships and measures including the optional relationships (Lab 6).

Below screenshot is how the auto-created report **may** look if you skipped creating the optional relationships and measures (Lab 6).
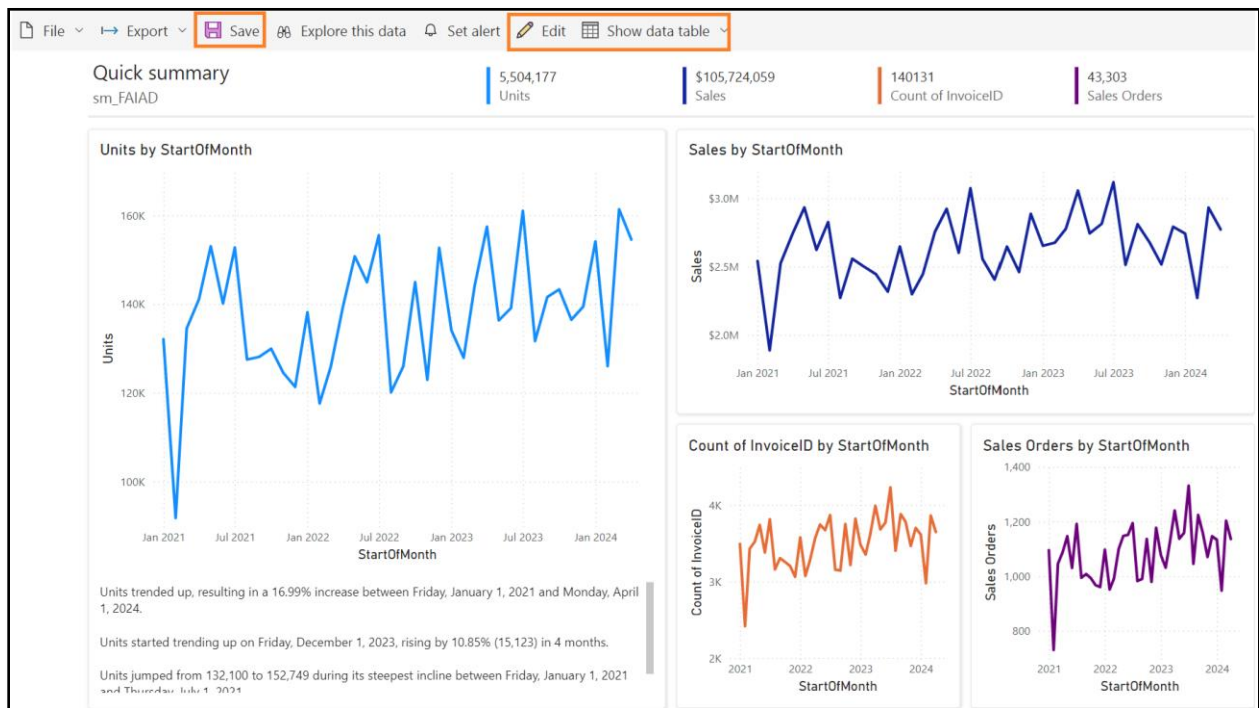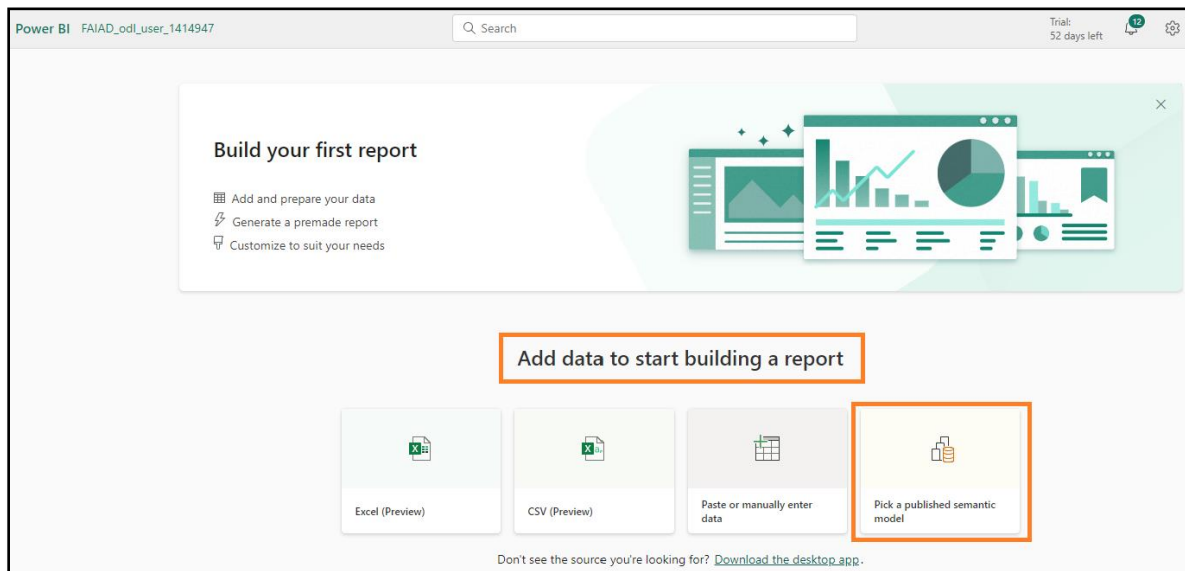
Task 2: Configure background for a New report

Let's create a new report using a blank canvas.

1. In the **left panel**, select your workspace name, **FAIAD_<username>** to be navigated to the workspace.
2. From the top menu, select **New -> Report**. You will be navigated to build your first report page.
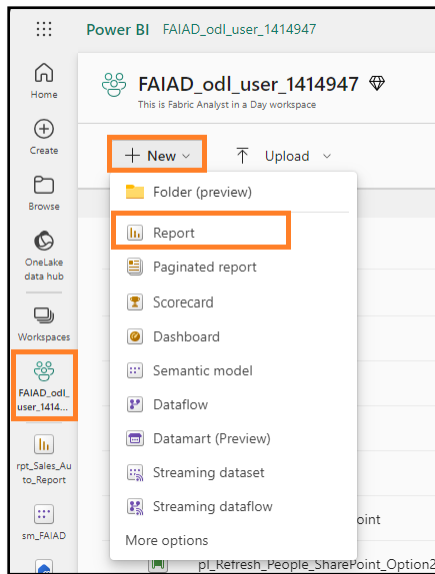
3. Select **Pick a published semantic model,** so we can pick the model we have created.


4. Pick a semantic model to use in your report dialog opens. Select **sm_FAIAD**.
5. Click the **arrow next to Auto-create report button**. Select **Create a blank report.** You will be navigated to a report page which looks similar to the Power BI Desktop report page.
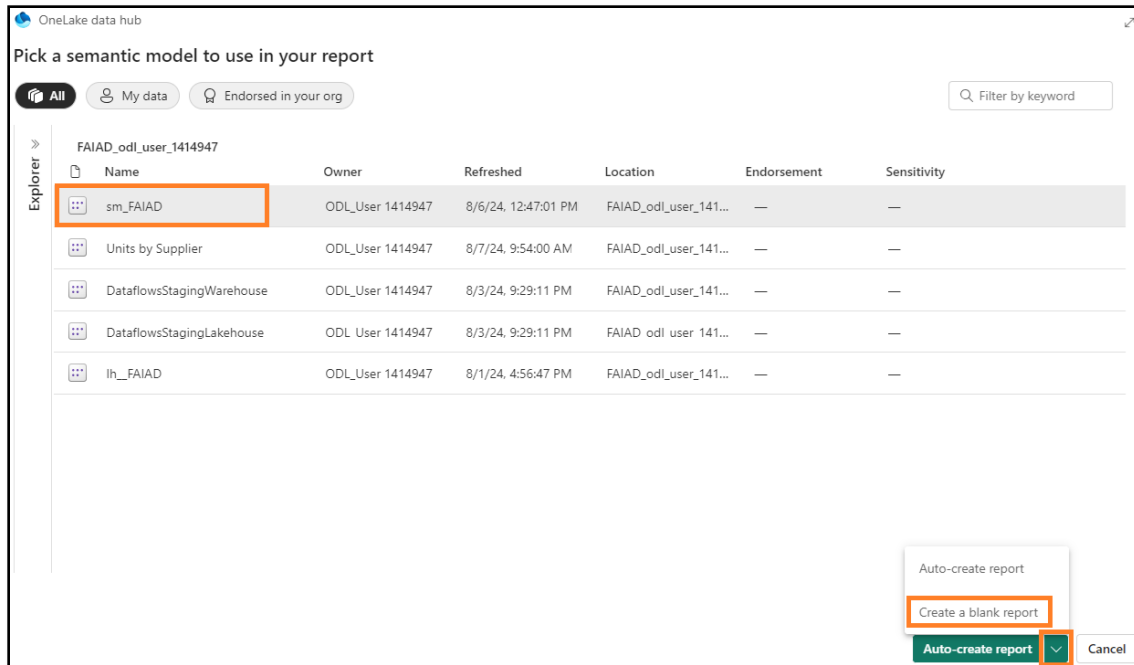
6. If you have not already opened it, open the **FAIAD.pbix** located **Reports** folder on the **desktop** of your lab environment.

We are going to use this report as a reference. We will start by adding the canvas background. We will create the report header, add a couple of KPIs, and create the Sales over time line chart. In the interest of time and with the understanding that you have experience with building visuals in Power BI Desktop, we will not be creating all the visuals.
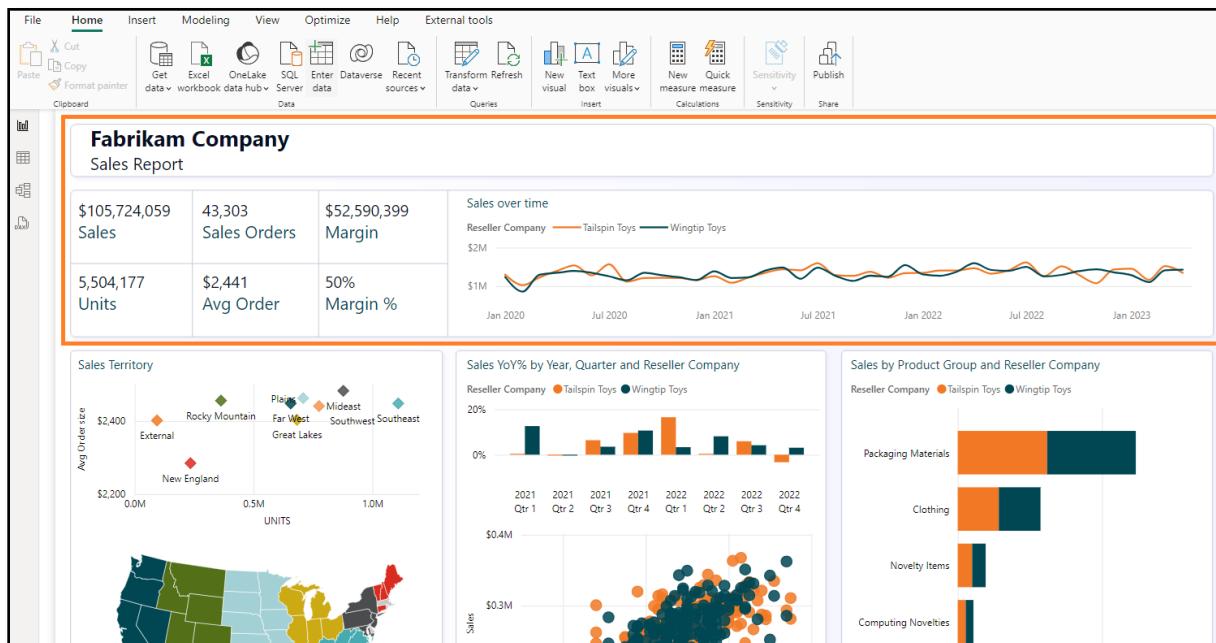
7. Navigate back to **Power BI canvas** in your browser.

8. Select **Format page icon** in **Visualization** pane.
9. Expand **Canvas background section**.
10. Select **Browse** from **Image** option. File explorer dialog opens.
11. Navigate to **Reports** folder on the **desktop** of your lab environment.
12. Select **Summary Background.png.**
13. Set **Image fit** dropdown to **Fit**.
14. Set Transparency to **0%**.
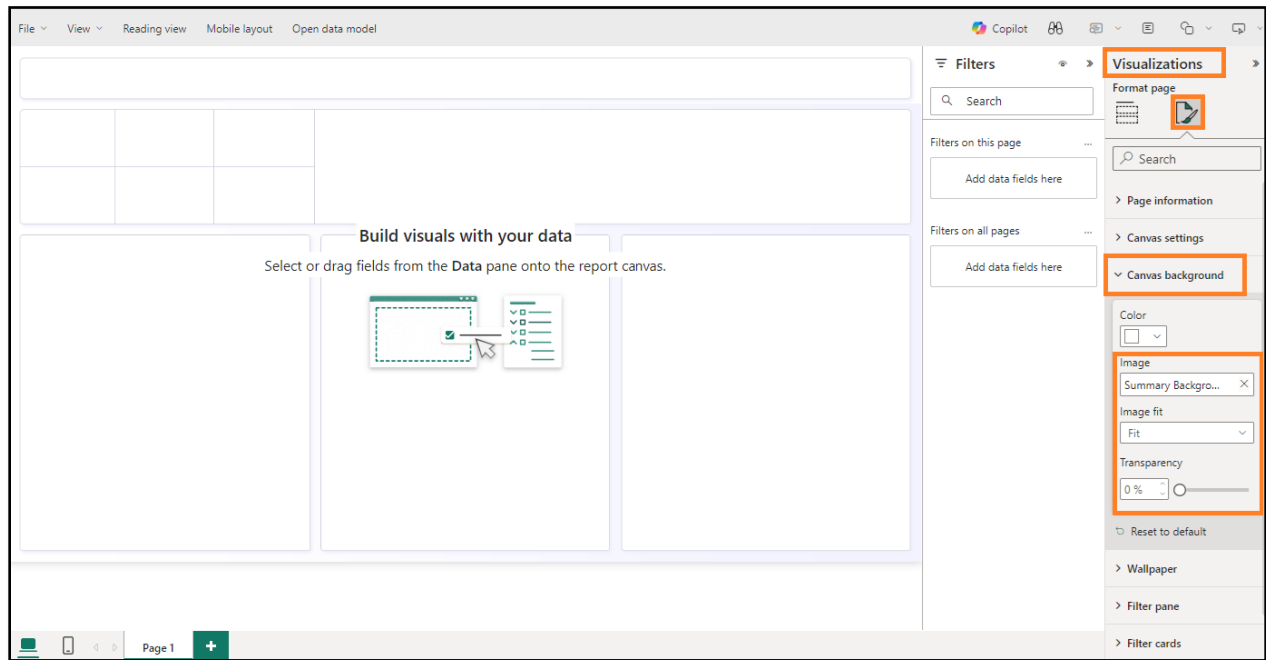
Task 3: Add Header to the report

1. Let's add the header in the top margin. From the **menu**, select **Text box**.
2. Enter **Fabrikam Company** as the first line in the text box.
3. Enter **Sales Report** as the second line in the text box.
4. Highlight **Fabrikam Company** and set **Font** to **Segoe UI** and **font size** to **18, bold**.
5. Highlight **Sales Report** and set **Font** to **Segoe UI** and **font size** to **14.**
6. With the **text box selected**, in the Format text box pane on the right, **expand Effects**.
7. Use **Background** slider to set it to **Off**.
8. Resize the **text box to fit in the top margin**.
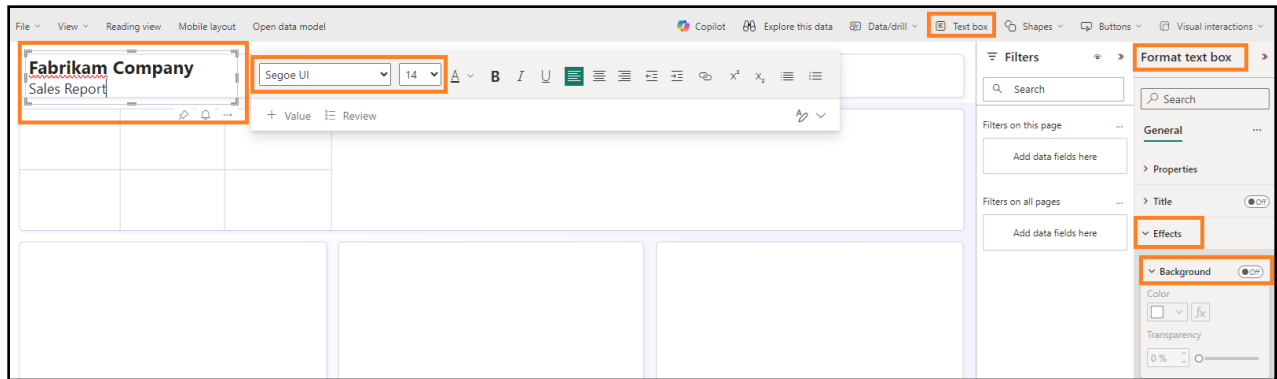
Task 4: Add KPIs to the report

1. Let's add Sales KPI. Select the **white space** in the canvas to take focus off the text box.
2. From the **Visualizations section** select **Multi-row card visual**.
3. From the **Data section** expand **Sales table**.
4. Select **Sales measure**.
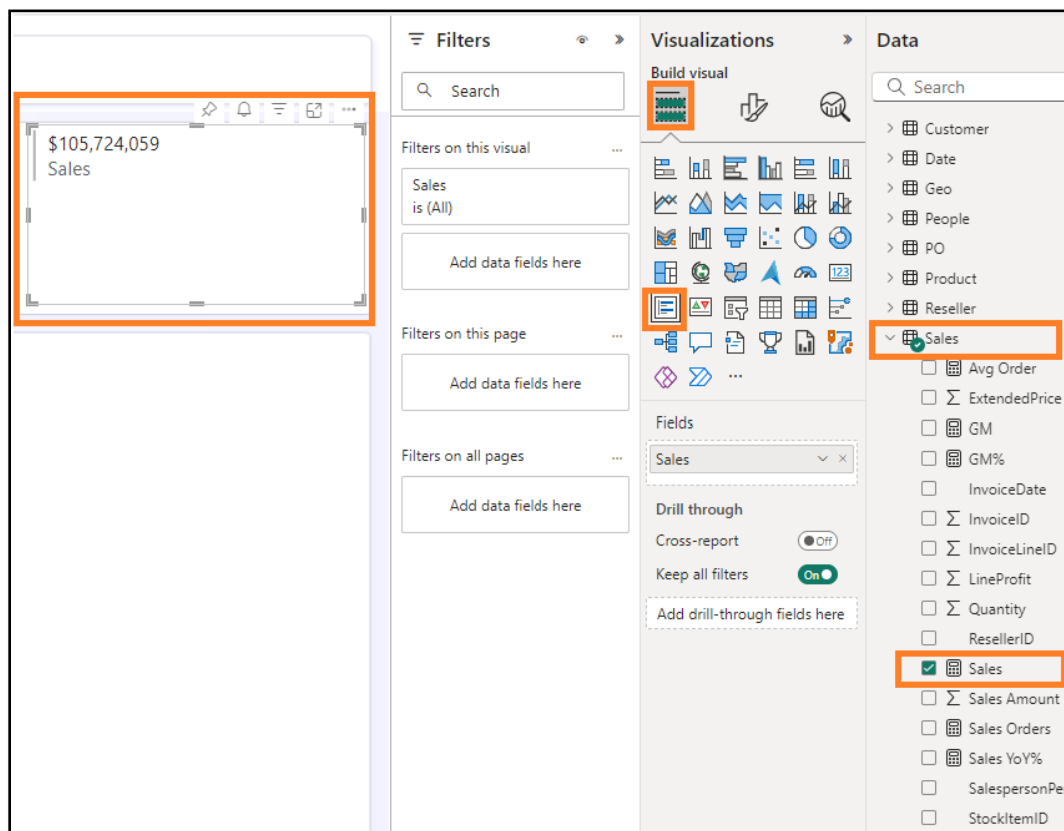
5. With **multi-row card visual selected**, select **Format visual icon** from **Visualizations** section.
6. Expand **Category labels** section.
7. Increase **font size** to **14**.
8. Select **Color drop down**. Color palette dialog opens.

9. Select **More Colors**.
10. Set Hex value to **#004753**.
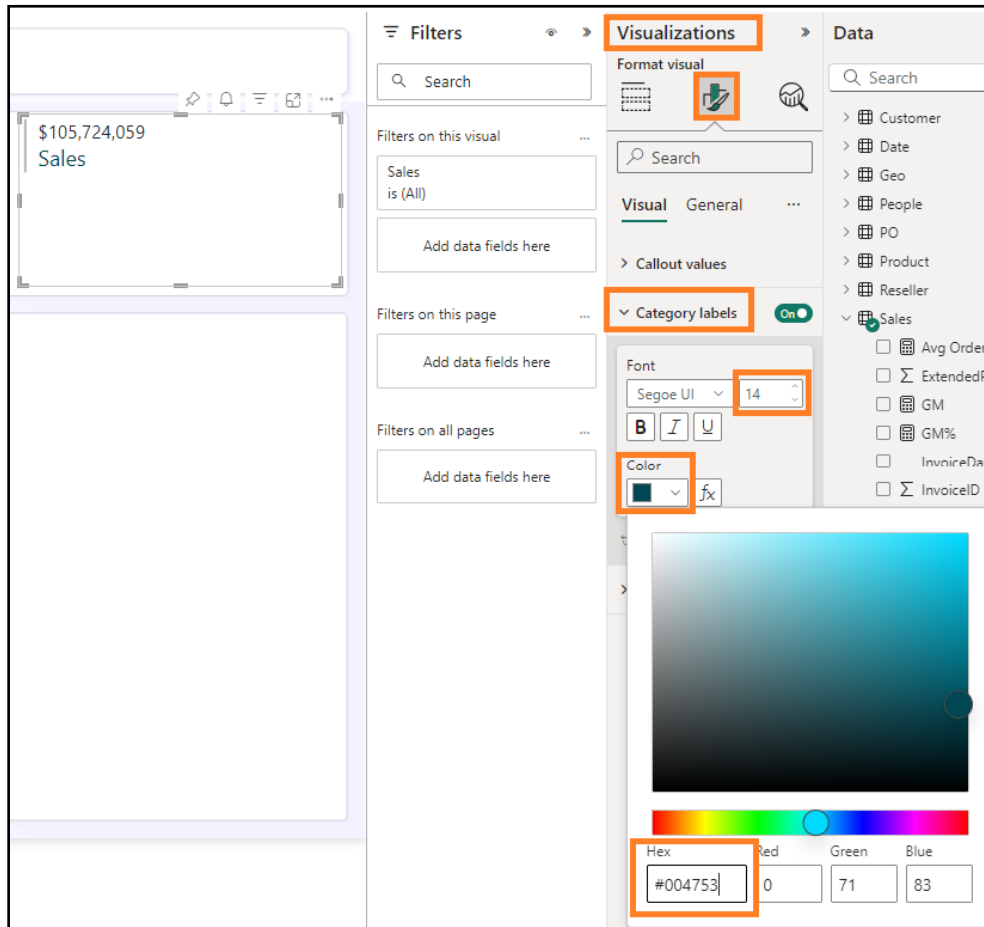
11. Expand **Cards** section.
12. Use **Accent bar** slider to set it to **Off**.

13. Select **General** in the Visualizations pane.
14. Expand **Effects section**.
15. Use **Background** slider to set it to **Off**.
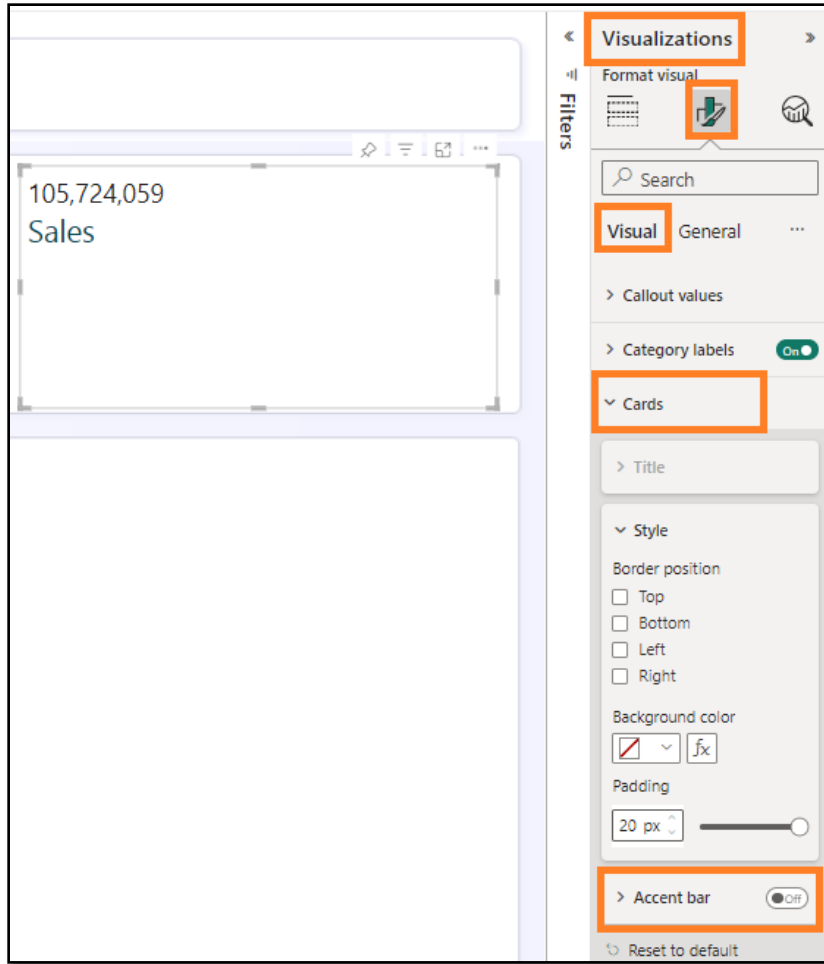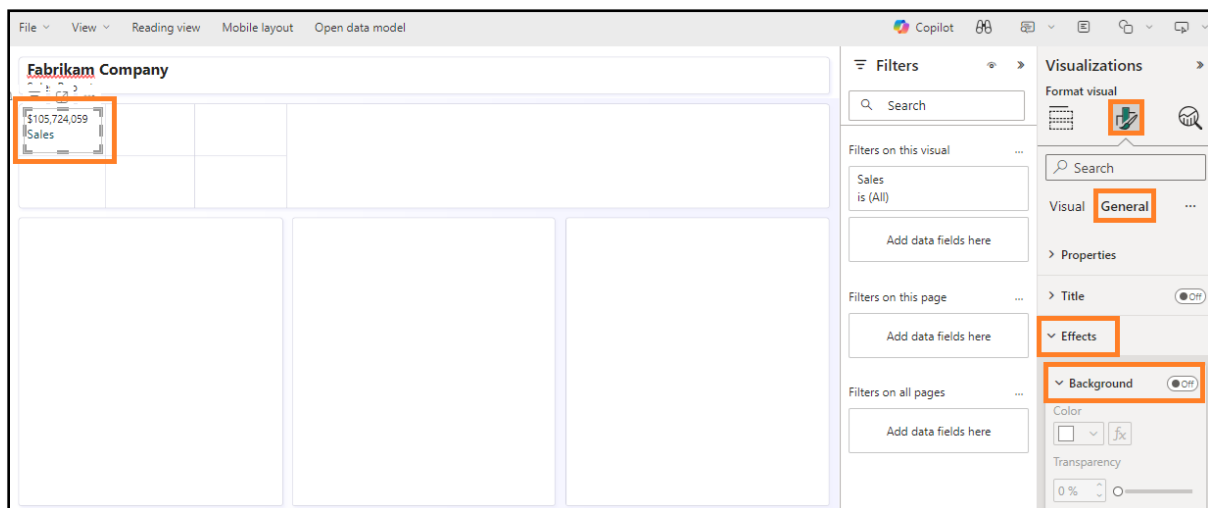16. Resize the **visual** and move it to the **left box as shown in the screenshot**.

17. Let's add another KPI. Select the **Sales multi-row card** we just created. **Copy** the visual by selecting **Ctrl+C** from your keyboard.
18. **Paste** the visual by selecting **Ctrl+V** from your keyboard. Notice the visual is pasted onto the canvas.
19. With the **new visual highlighted**, in the **Visualization pane -> Build visual -> Fields** section remove **Sales** measure.
20. From the **Data** section, expand **Sales** table and select **Units** measure.
21. Resize the **visual** and **place it in the box below the Sales visual**.
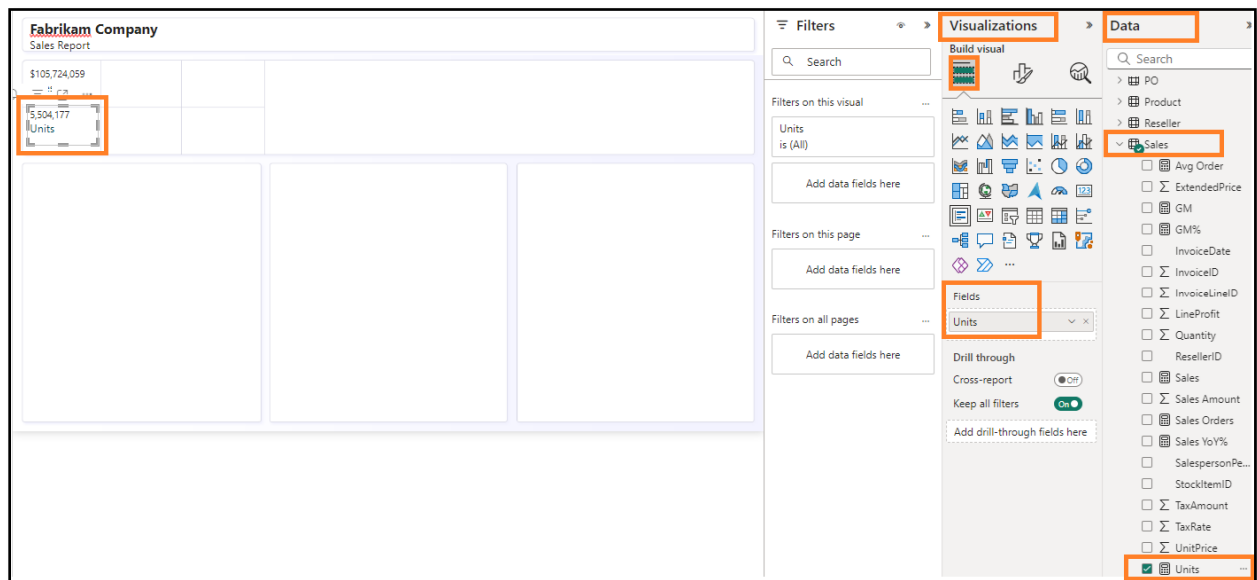
Task 5: Add Line chart to the report

Let's create a line chart to visualize Sales over time by Reseller Company.

1. Select the **white space** in the canvas to take focus off the multi-row card visual.
2. From the **Visualizations section** select **Line chart**.
3. From the **Data section** expand **Date** table.
4. Select **Year** field. Notice Year is a summed by default and added to the Y-axis. Let's rectify this.
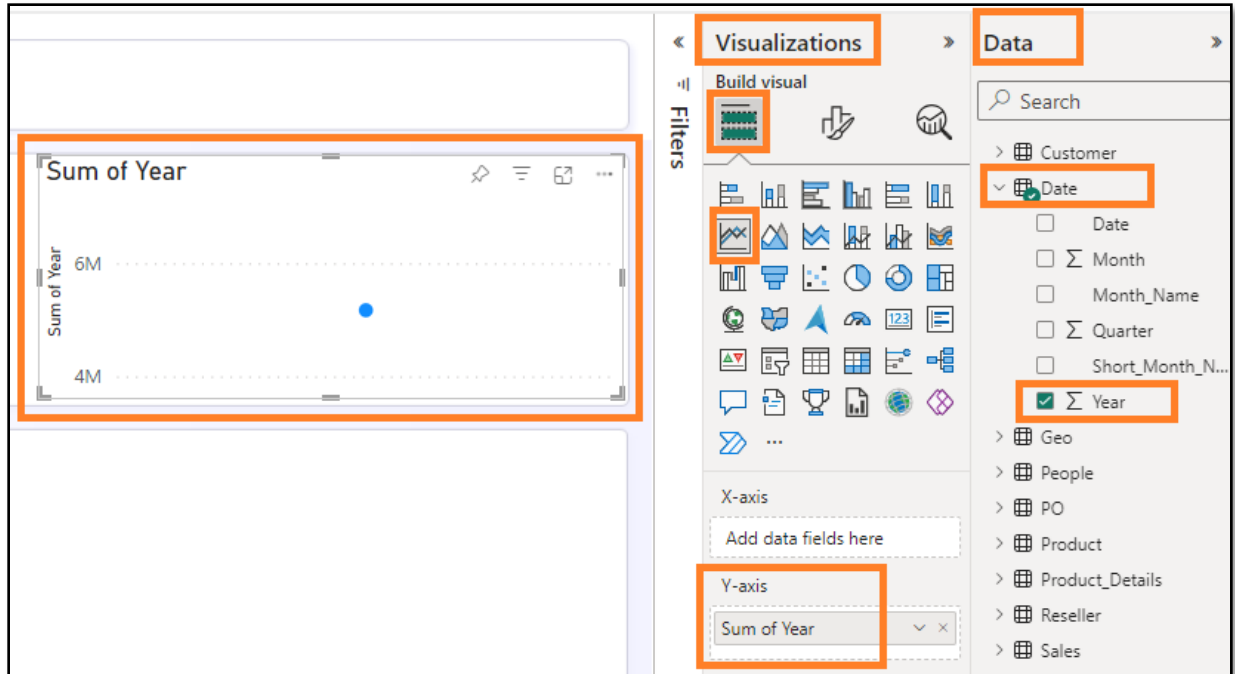
Task 6: Save the report

Let's save the report before we navigate away from the report to make changes to the model.

1. From the menu select **File -> Save**.
2. Save your report dialog opens. Name the report as **rpt_Sales_Report**
      Note: We are prefixing report name with rpt which is short for report.
3. Make sure the report is saved in **FAIAD_<username>** workspace**.**
4. Select **Save.** Notice the report is saved and you are in view mode.
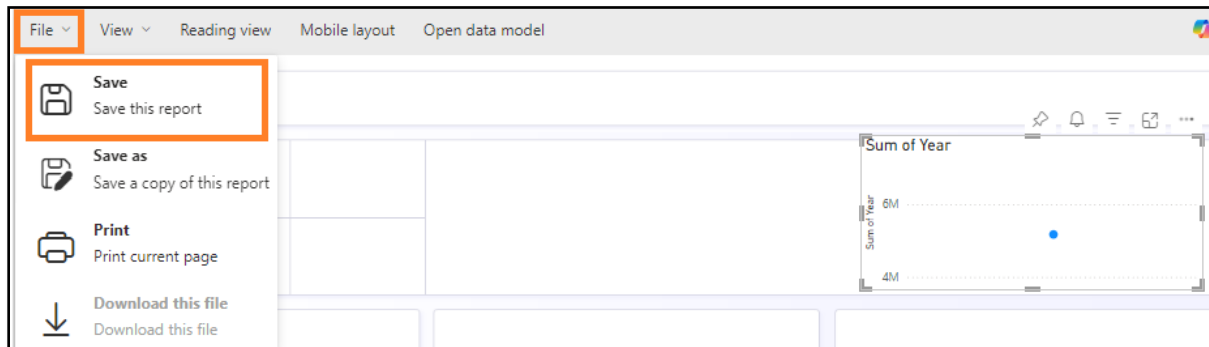
Task 7: Configure Year column in Date table

1. From the **top menu**, select **Edit** to go back into Edit mode.
2. From the **top menu**, select **Open data model**. Notice semantic model is opened in a new browser window/tab.
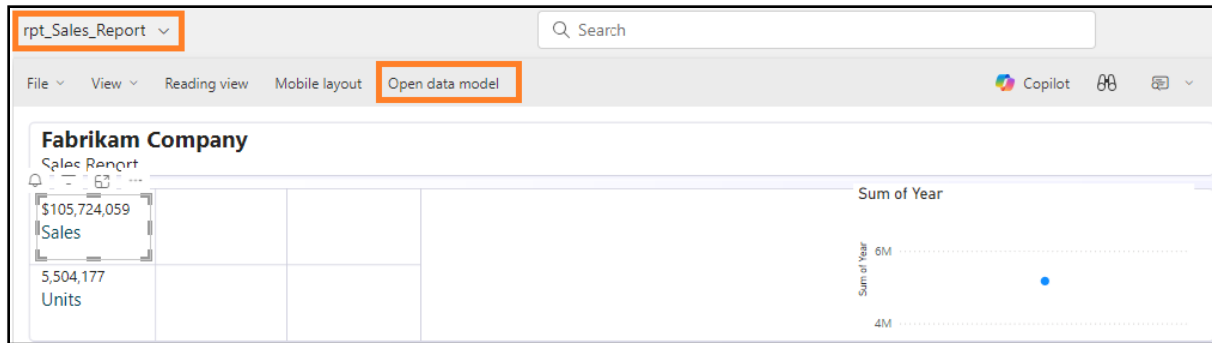
3. From the **Data panel on the right,** select Tables.

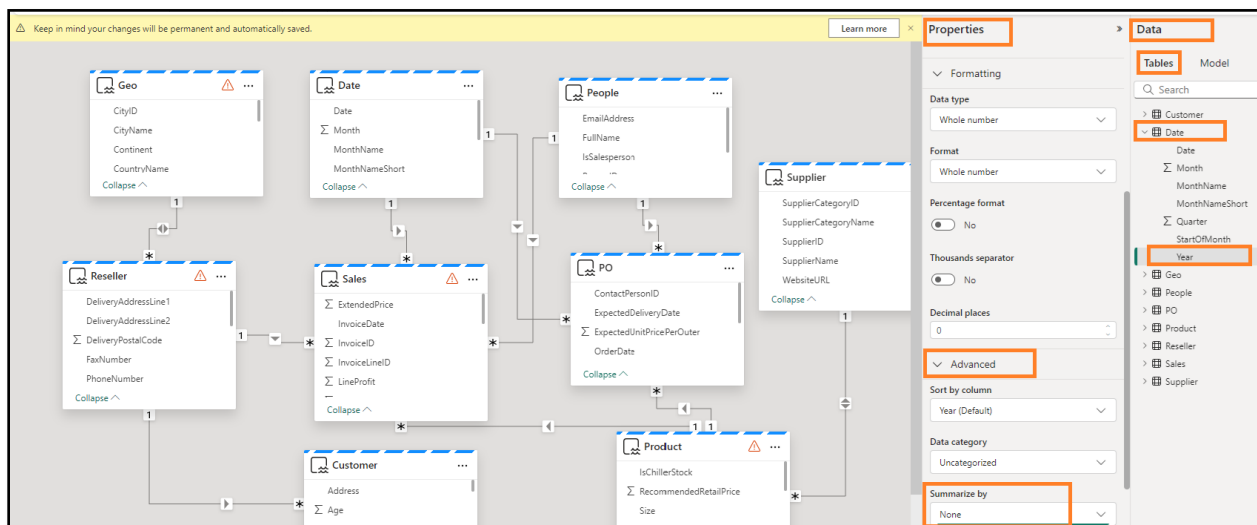4. Expand **Date** table.

5. Select **Year** column.

6. In the **Properties** pane on the right, expand **Advanced** section.

7. In the **Summarize by** drop down select **None**.

8. Navigate back to **report window/tab** of the browser.

9. On the **Data pane** of the right, expand **Date** table. Notice Year is not a summation field.

10. With the **Line chart visual selected**, **remove Sum of Year** from the Y-axis.

11. Select **Year** field and it will be added to the **X-axis**.

12. Expand **Sales** table and select **Sales measure**.
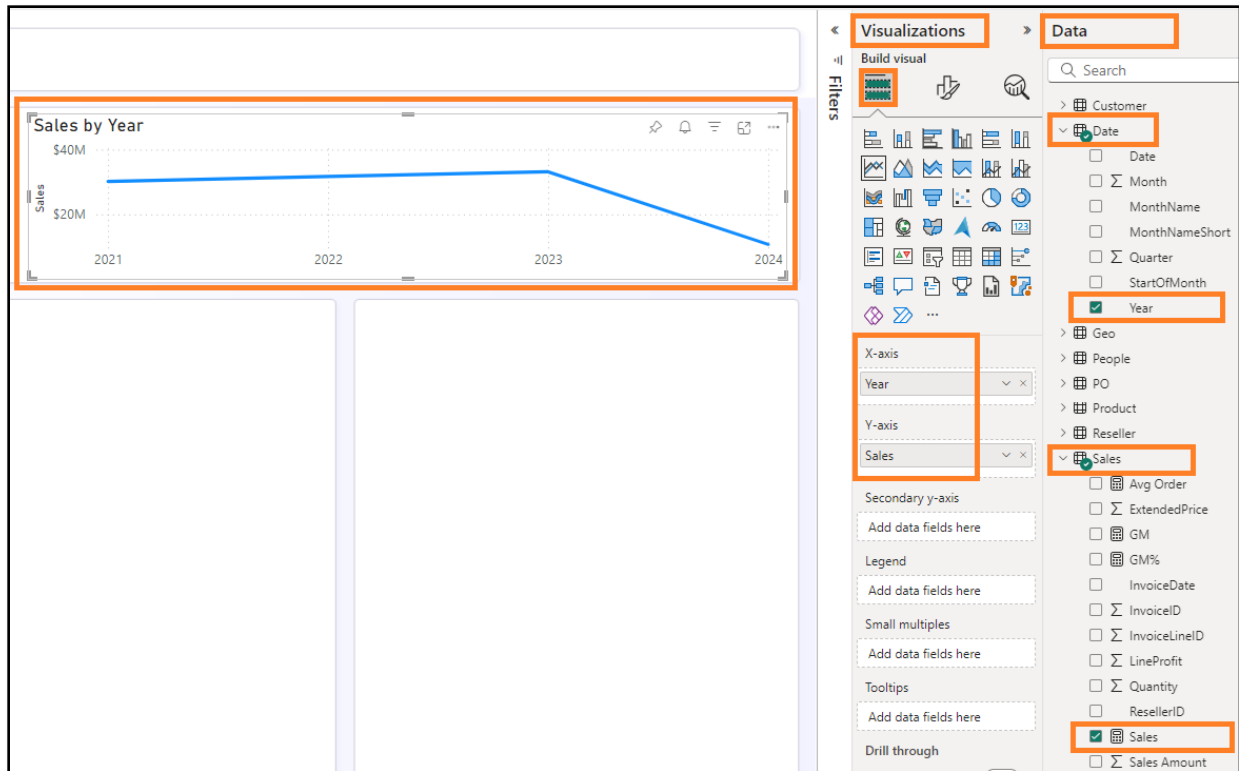
Task 8: Configure Month Name column in Date table
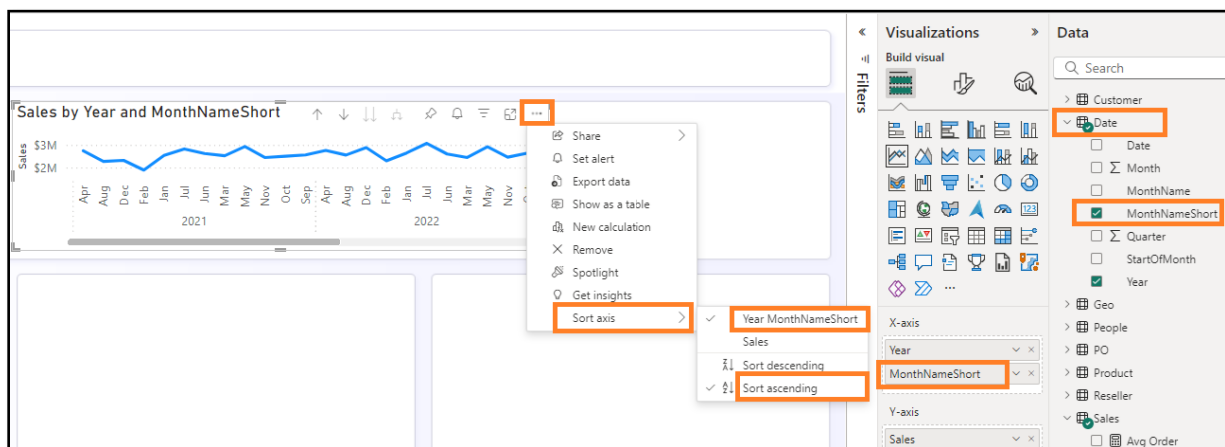
1. Let's add Month to this chart. From the Date table, drag **MonthNameShort** field below **Year** in the **X-axis**. Notice the visual is sorted by Sales. Let's sort it by **MonthNameShort**.
2. Select the **ellipsis (…)** on the top right corner of the visual.
3. Select **Sort axis -> Year Short_Month_Name**.
4. Select the **ellipsis (…)** on the top right corner of the visual.
5. Select **Sort axis -> Sort ascending**.

Note: The months are sorted alphabetically. Let's fix this.
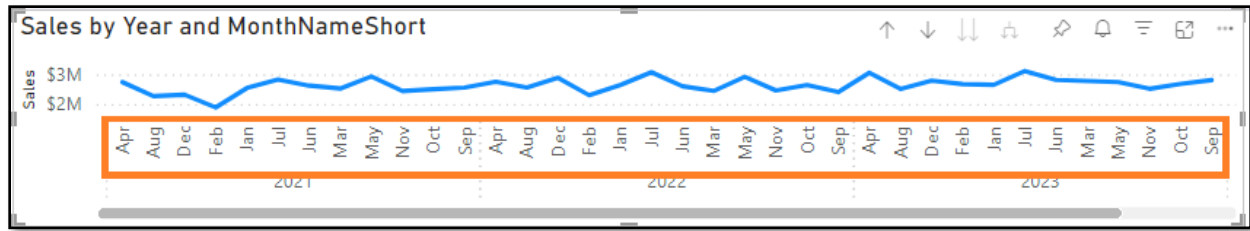
6. Navigate back to the **browser window/tab** where you have the semantic model open.
7. In the **Data** pane, expand **Date** table.
8. Select **MonthNameShort** column.
9. In the **Properties** pane on the right, expand **Advanced** section.
10. In the **Sort by column** drop down select **Month**.
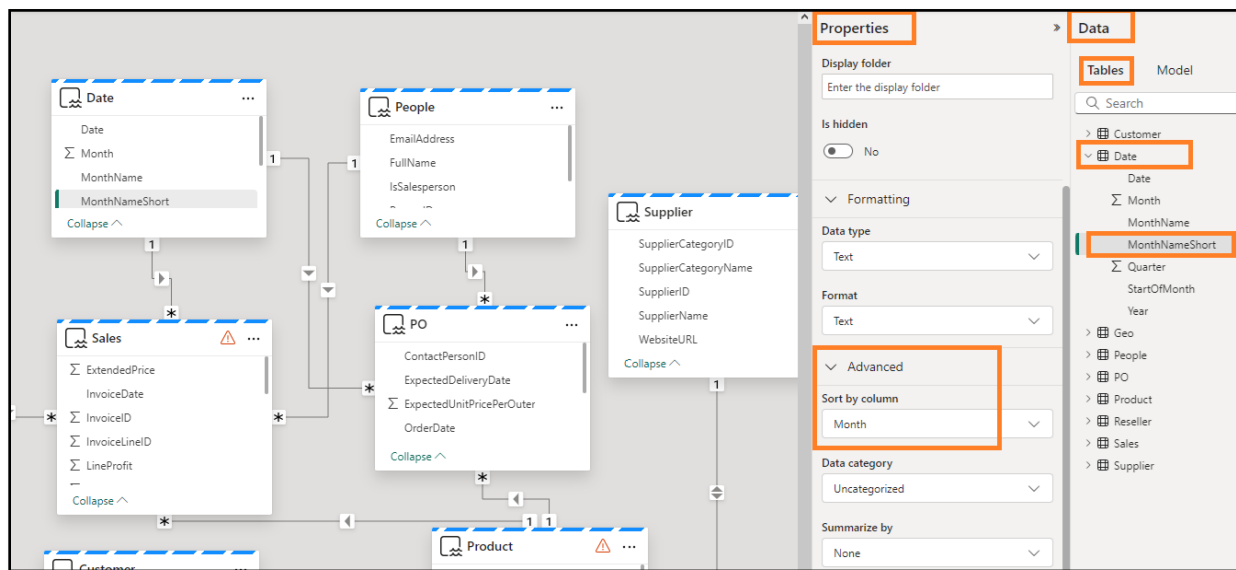
11. Navigate back to **report window/tab** of the browser. Notice now months are sorted properly.
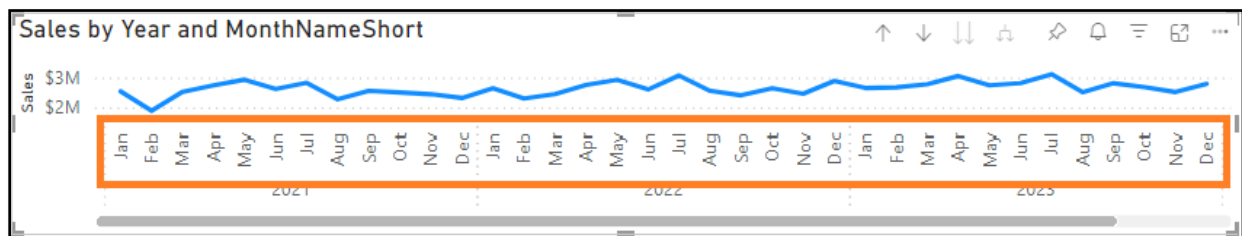
Task 9: Format Line chart

Notice how easy it is to update the semantic model while building the reports. This gives a seamless interaction like Power BI Desktop.

5. With the **Line chart visual selected**, in the **Data section** expand **Reseller** table.
6. Drag **Reseller -> Reseller Company** field to the **Legend** section.
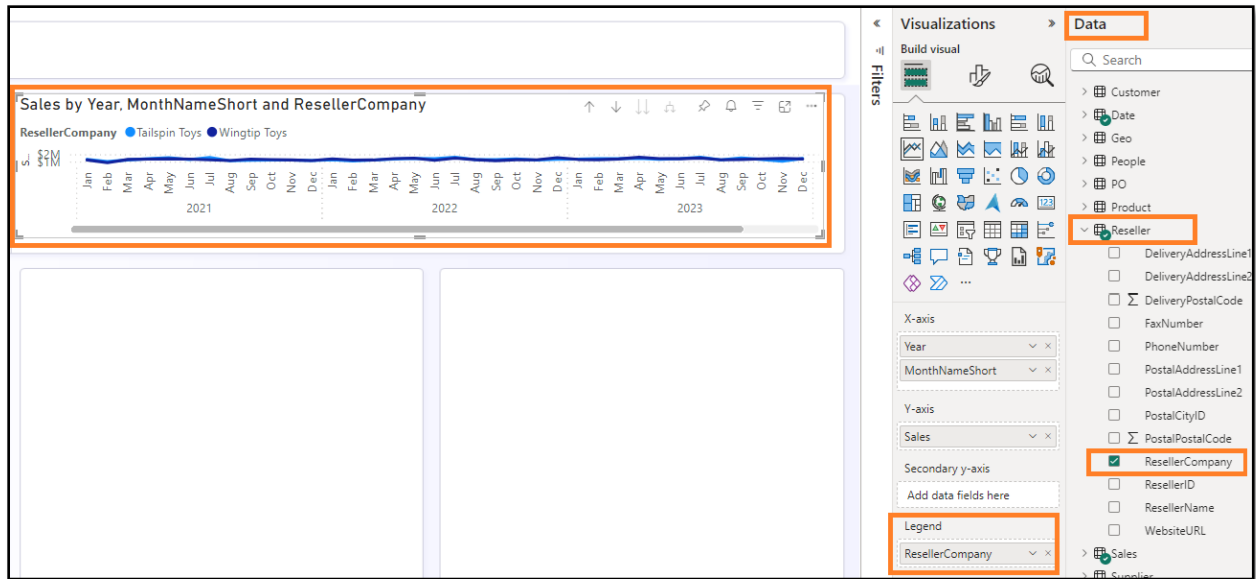
7. With the **Line chart visual selected**, from the **Visualization** section select **Format visual icon -> General**.
8. Expand **Title** section.
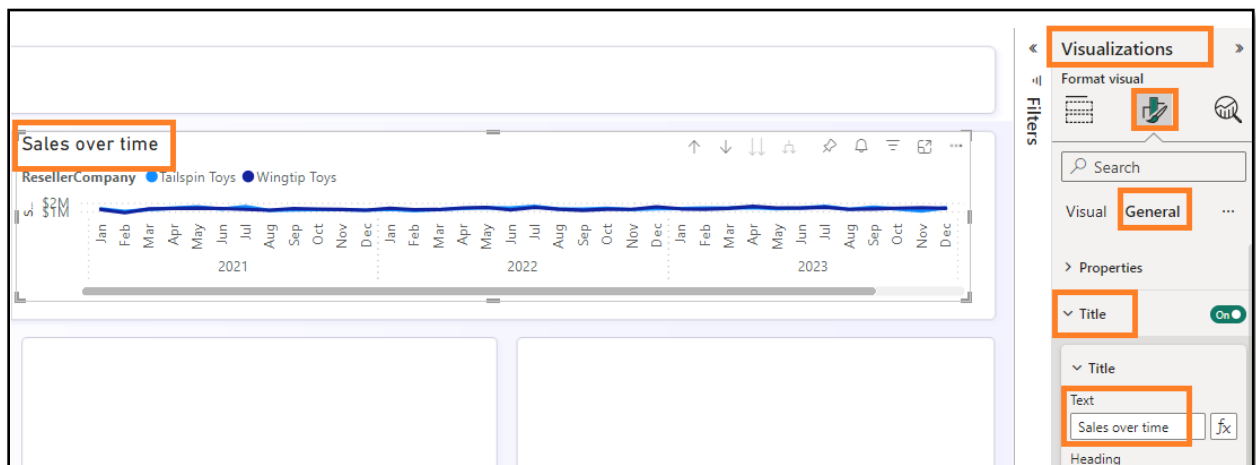9. Set **Title** text to **Sales over time**.
10. Expand **Effects** section.
11. Use **Background** slider to set it to **Off**.

12. From the **Visualization** section select **Format visual icon -> Visual**.
13. Expand **Lines** section.
14. In the **Apply settings to -> Series dropdown,** select **Tailspin Toys.**
15. Expand **Colors** section.
16. Set **color** to **#F17925**
17. In the **Apply settings to -> Series dropdown,** select **Wingtip Toys.**
18. Set **color** to **#004753**
19. Resize the **visual** and move it to the **top right box as shown in the screenshot.**

20. Scroll to the right on the visual and **notice we have data through April 2024**.
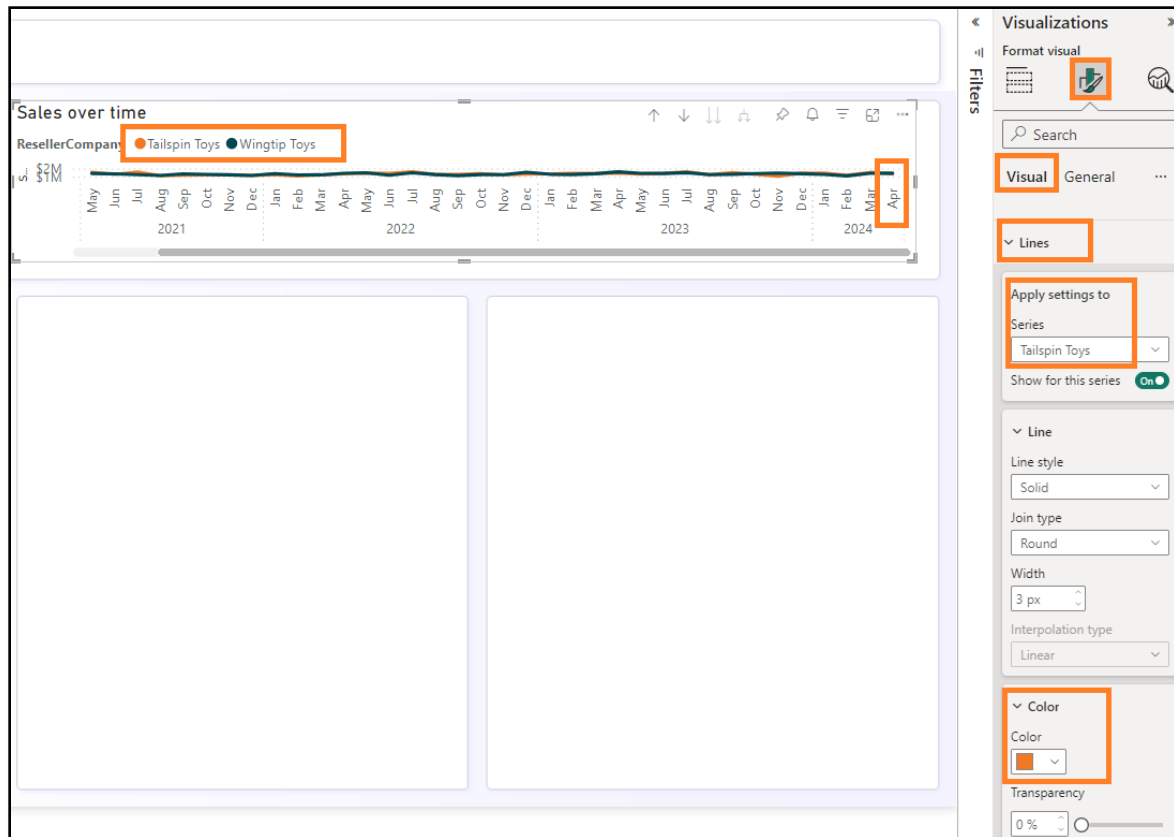
21. Let's save the report, from the menu select **File -> Save**.

As mentioned earlier, we will not build all the visuals in this lab. At your leisure, feel free to build more visuals.

Task 10: Connect Power BI Desktop to Semantic model
Now let's see how easy it is to connect Power BI Desktop to the semantic model and build visuals.

1. Open the **FAIADTemplate.pbix** located **Reports** folder on the **desktop** of your lab environment.
2. From the ribbon select **Home -> OneLake data hub -> Power BI semantic models**.
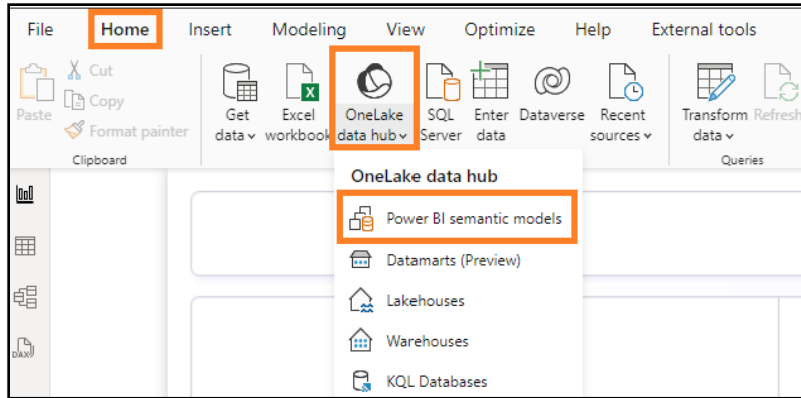
3. OneLake data hub dialog opens. Select **sm_FAIAD**, the semantic model we have created.

4. Select **Connect**. Notice in the Data pane, we have the tables from the semantic model.
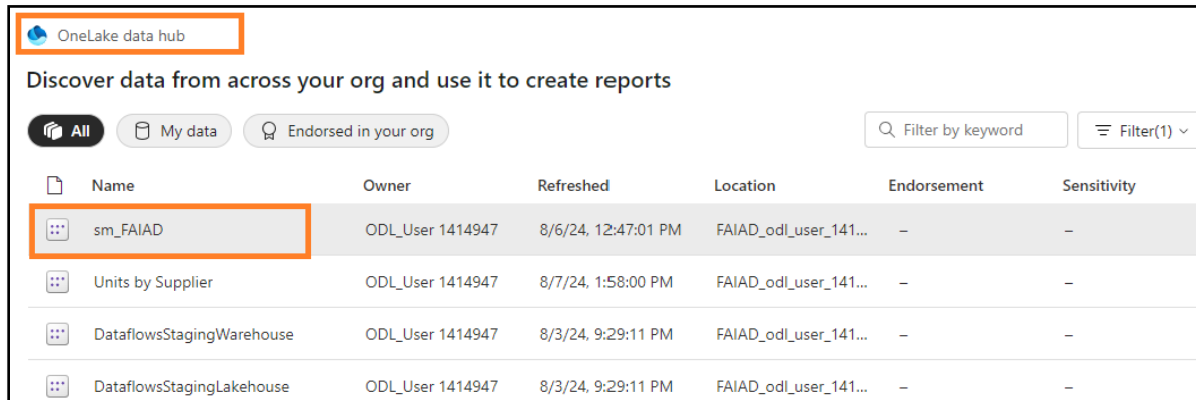
5. From the **left panel**, select **Model view**. Notice we can view the relationship between tables.
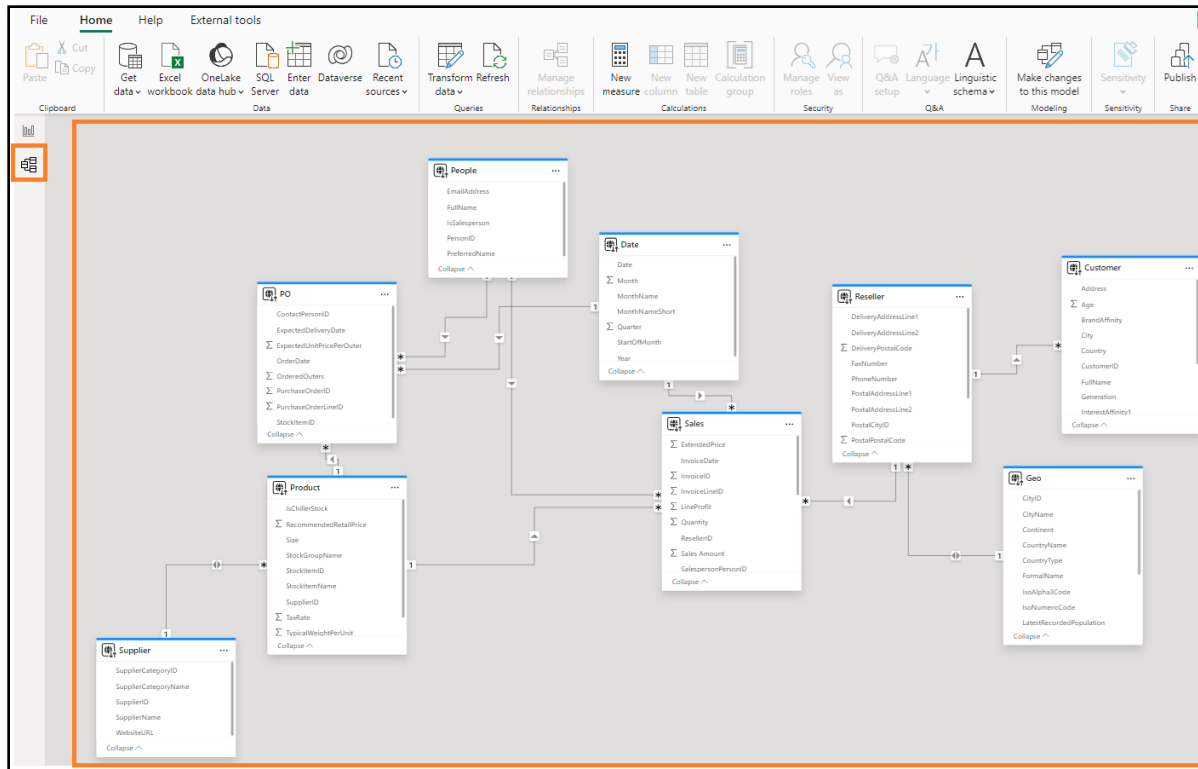
6. From the **left panel**, select **Report view** to navigate back to the Report view.
7. If you have not already done so, open the **FAIAD.pbix** located **Reports** folder on the **desktop** of your lab environment.
8. Select the **report title visual**.
9. From the ribbon, select **Home -> Copy**.

10. Navigate to **FAIADTemplate.pbix** and select the report canvas.
11. From the ribbon, select **Home -> Paste**.
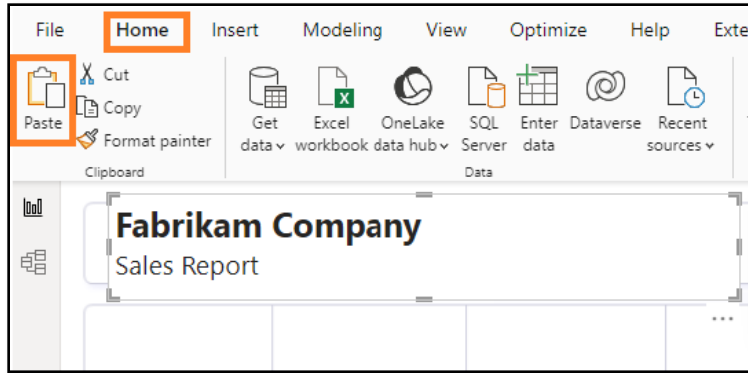
12. Similarly copy and paste the **Sales and Units KPIs**. FYI – multiple visuals can be copied and pasted together.

Notice it is easy to copy visuals from an existing report and paste it to a report that connects to semantic model. Note that the table names, column names, measure names must be the same for copy and paste to work. If they are not the same you may have an error, but this can be easily resolved.

13. Navigate to **FAIAD.pbix** and select Sales over time line chart.
14. From the ribbon, select **Home -> Copy**.
15. Navigate to **FAIADTemplate.pbix** and select the report canvas.
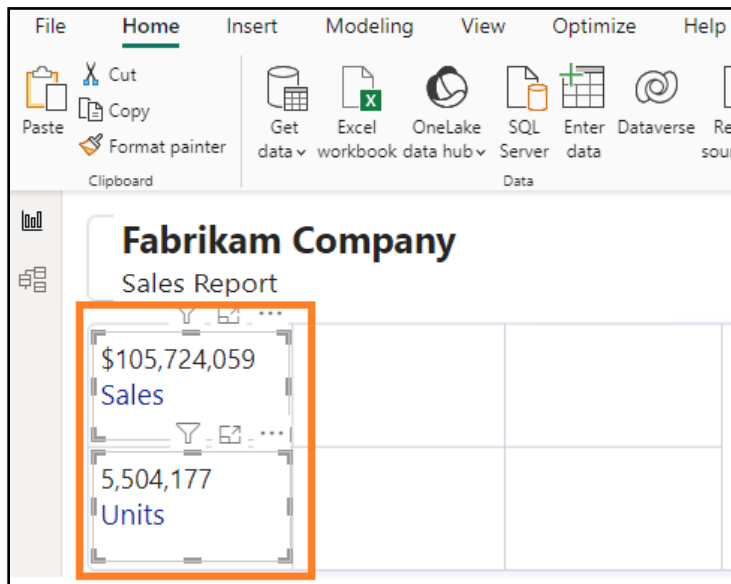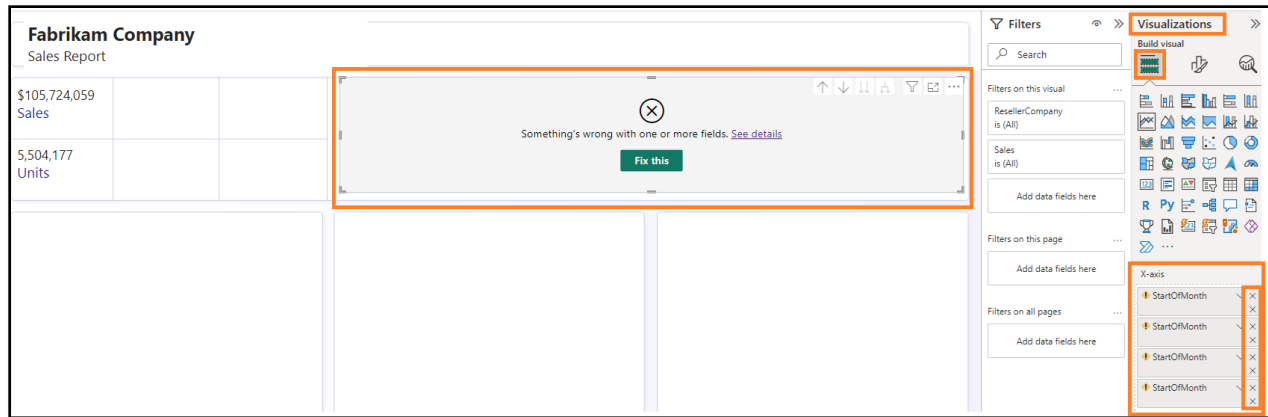16. From the ribbon, select **Home -> Paste**. Notice that the visual does not render. This is because currently semantic model does not create hierarchy from date field.
17. Let's fix this. In **Visualization** panel, under **X-axis** delete **StartOfMonth**.

18. From the **Data pane**, expand **Date** table.

19. Drag **StartOfMonth** field into **X-axis**. This fixes the visual. You may have to format the visual.
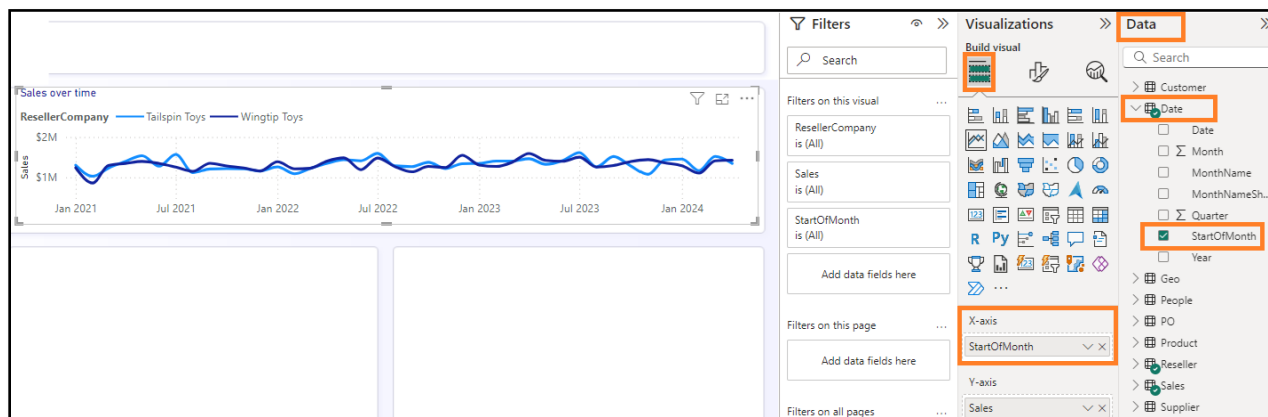
20. Let's save the report, from the ribbon select **File -> Save**.

Task 11: Add new data to simulate Direct Lake Mode

Typically, in Import mode, once data in the source is refreshed, we need to refresh the Power BI model after which the data in the report is updated. With Direct Query mode, once data is refreshed in source, it is available in Power BI report. However direct query mode is typically slow. To solve this problem, Microsoft Fabric has introduced Direct Lake mode. Direct Lake is a fast path to load the data from the lake straight into the Power BI engine, ready for analysis.

Let's explore the scenario where data is updated in the ADLS Gen2 and the changes are immediately reflected in Power BI report without running any refreshes.

In a real scenario, data is updated at the source. Since we are in a training environment, we will simulate this. We have Sales data through April 2024. Let's add Sales data for May 2024 by creating a shortcut to the May 2024 file in ADLS Gen2 and updating the Sales view.

1. Navigate back to the **browser**.

2. Select **FAIAD_<username>** from the left menu bar to navigate to workspace home.

3. Select **lh_FAIAD** to navigate into the Lakehouse.
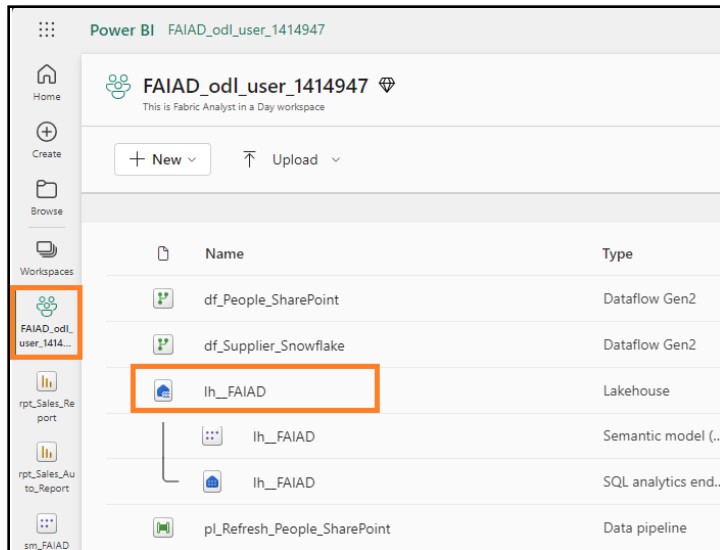
4. From the **Explorer pane** on the left, select the **ellipsis** next to **Tables**.

5. Select **New shortcut**.
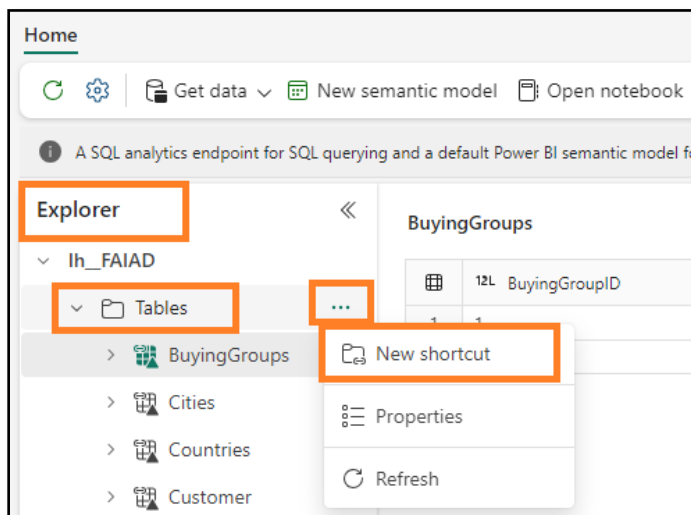
6. New shortcut dialog opens. Under **External sources**, select **Azure Data Lake Storage Gen2**.
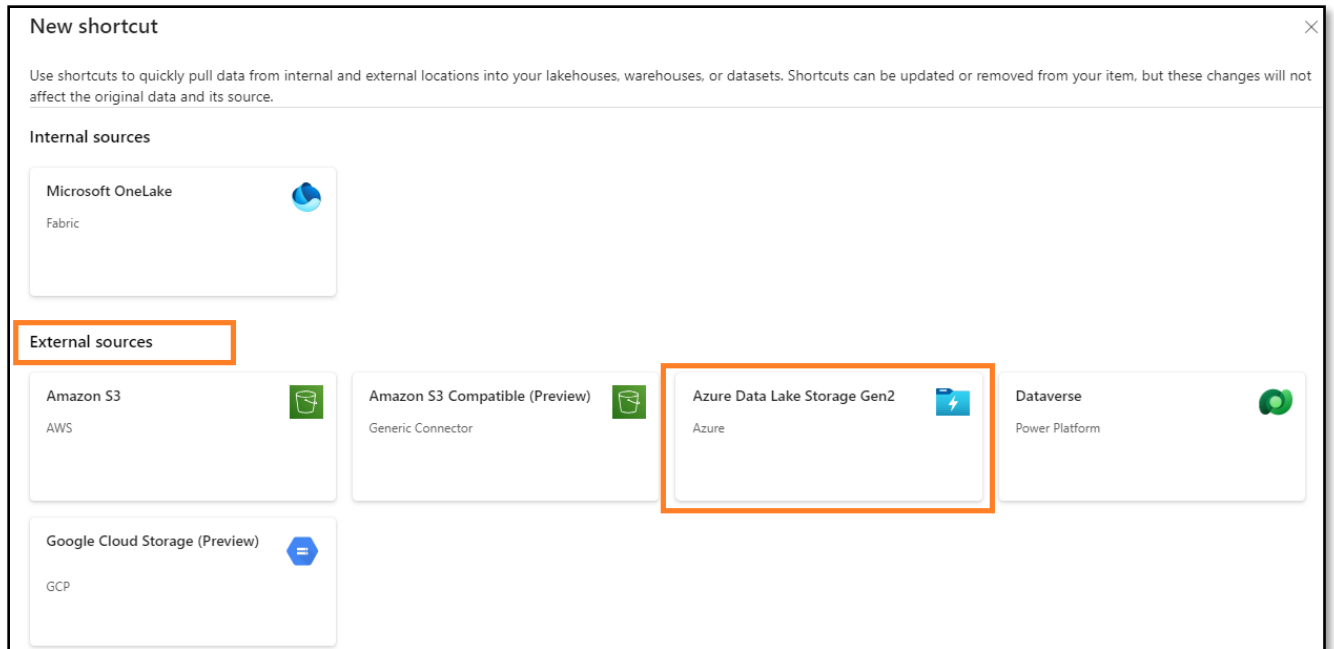
7. Under **Connection Settings -> URL** enter this link
   https://stvnextblobstorage.dfs.core.windows.net/fabrikam-sales
8. Since we have created the connection earlier, it is recognized. Select **Next**.
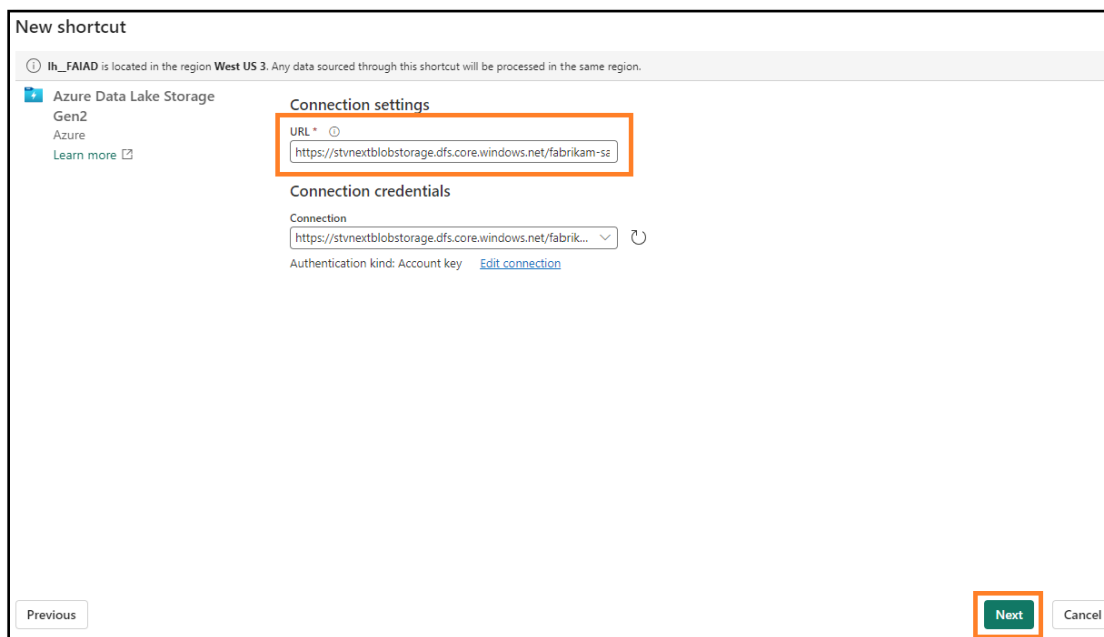
9. You will be connected to ADLS Gen2 with the directory structure displayed in the left panel. Expand **Delta-Parquet-Format-FY25.**
10. Select **Sales.Invoices_May.**
11. Select **Next**.
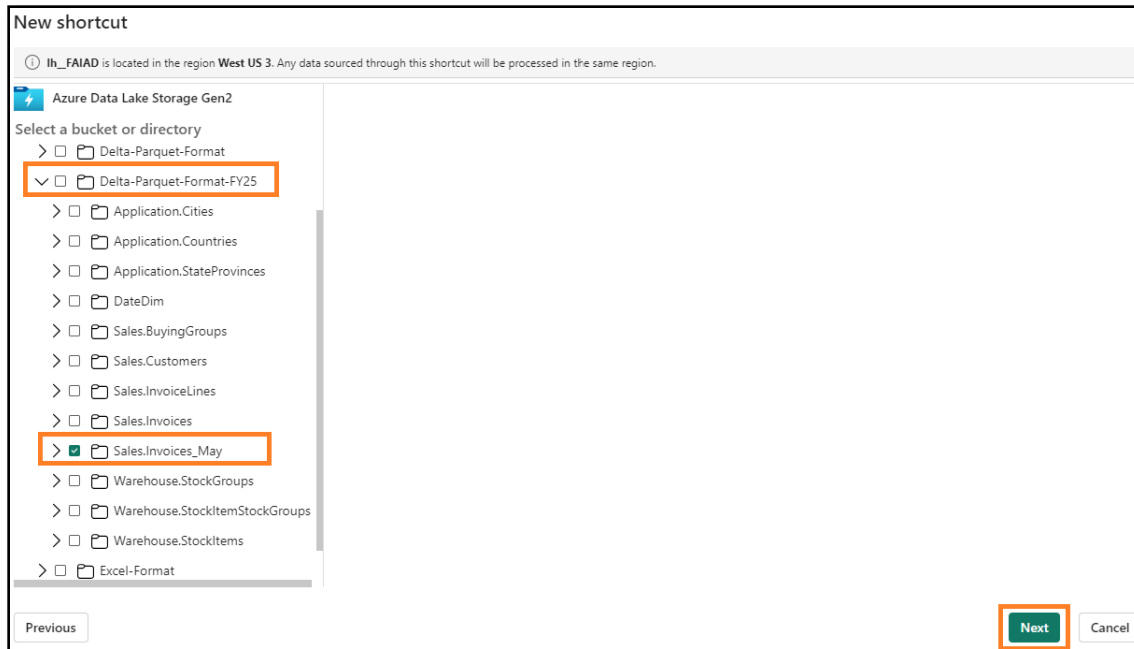
12. You will be navigated to the next dialog where we can edit the names. Select the **Edit icon** under Actions for **Sales.Invoices_May**.

13. Rename **Sales.Invoices_May to InvoicesMay.**

14. Select the **check mark** next to the name to save the change.

15. Select **Create**.

Notice in the **Explorer pane** on the left, we have InvoicesMay table. Now we need to update Sales view.

16. On the **top right** of the screen, select **Lakehouse -> SQL analytics endpoint**.

17. From the top menu, select **Home -> New SQL query**. A new SQL query pane opens.
18. **Copy** the below code and **paste** it into the SQL query pane**.**

```sql
ALTER VIEW [dbo].[Sales] AS (
select [$Outer].[InvoiceLineID] as [InvoiceLineID],
    [$Outer].[InvoiceID] as [InvoiceID],
    [$Outer].[StockItemID] as [StockItemID],
    [$Outer].[Quantity] as [Quantity],
    [$Outer].[UnitPrice] as [UnitPrice],
    [$Outer].[TaxRate] as [TaxRate],
    [$Outer].[TaxAmount] as [TaxAmount],
    [$Outer].[LineProfit] as [LineProfit],
    [$Outer].[ExtendedPrice] as [ExtendedPrice],
    [$Outer].[CustomerID] as [ResellerID],
    [$Outer].[SalespersonPersonID] as [SalespersonPersonID],
    [$Outer].[InvoiceDate] as [InvoiceDate],
    [$Outer].[t0_0] as [Sales Amount]
from
(
    select [_].[InvoiceLineID] as [InvoiceLineID],
        [_].[InvoiceID] as [InvoiceID],
        [_].[StockItemID] as [StockItemID],
        [_].[Quantity] as [Quantity],
        [_].[UnitPrice] as [UnitPrice],
        [_].[TaxRate] as [TaxRate],
        [_].[TaxAmount] as [TaxAmount],
        [_].[LineProfit] as [LineProfit],
        [_].[ExtendedPrice] as [ExtendedPrice],
        [_].[CustomerID] as [CustomerID],
        [_].[SalespersonPersonID] as [SalespersonPersonID],
        [_].[InvoiceDate] as [InvoiceDate],
        [_].[ExtendedPrice] - [_].[TaxAmount] as [t0_0]
    from
    (
        select [$Outer].[InvoiceLineID],
            [$Outer].[InvoiceID],
            [$Outer].[StockItemID],
            [$Outer].[Quantity],
```
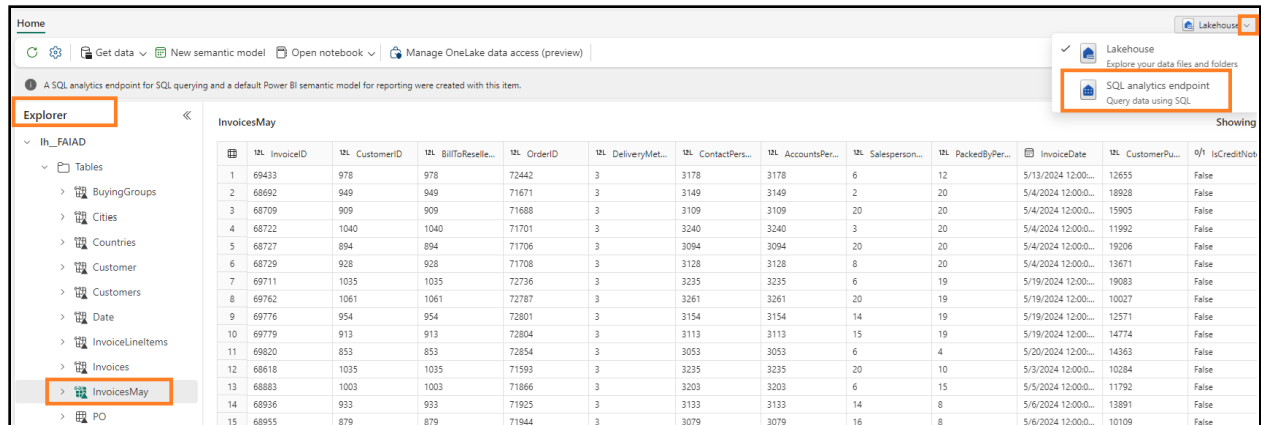
```
            [$Outer].[UnitPrice],
            [$Outer].[TaxRate],
            [$Outer].[TaxAmount],
            [$Outer].[LineProfit],
            [$Outer].[ExtendedPrice],
            [$Inner].[CustomerID],
            [$Inner].[SalespersonPersonID],
            [$Inner].[InvoiceDate]
        from [lh_FAIAD].[dbo].[InvoiceLineItems] as [$Outer]
        inner join
        (
            select [_].[InvoiceID] as [InvoiceID2],
                [_].[CustomerID] as [CustomerID],
                [_].[BillToResellerID] as [BillToResellerID],
                [_].[OrderID] as [OrderID],
                [_].[DeliveryMethodID] as [DeliveryMethodID],
                [_].[ContactPersonID] as [ContactPersonID],
                [_].[AccountsPersonID] as [AccountsPersonID],
                [_].[SalespersonPersonID] as [SalespersonPersonID],
                [_].[PackedByPersonID] as [PackedByPersonID],
                [_].[InvoiceDate] as [InvoiceDate],
                [_].[CustomerPurchaseOrderNumber] as [CustomerPurchaseOrderNumber],
                [_].[IsCreditNote] as [IsCreditNote],
                [_].[CreditNoteReason] as [CreditNoteReason],
                [_].[Comments] as [Comments],
                [_].[DeliveryInstructions] as [DeliveryInstructions],
                [_].[InternalComments] as [InternalComments],
                [_].[TotalDryItems] as [TotalDryItems],
                [_].[TotalChillerItems] as [TotalChillerItems],
                [_].[DeliveryRun] as [DeliveryRun],
                [_].[RunPosition] as [RunPosition],
                [_].[ReturnedDeliveryData] as [ReturnedDeliveryData],
                [_].[ConfirmedDeliveryTime] as [ConfirmedDeliveryTime],
                [_].[ConfirmedReceivedBy] as [ConfirmedReceivedBy],
                [_].[LastEditedBy] as [LastEditedBy2],
                [_].[LastEditedWhen] as [LastEditedWhen2]
            from
            (
                select [$Table].[InvoiceID] as [InvoiceID],
                    [$Table].[CustomerID] as [CustomerID],
                    [$Table].[BillToResellerID] as [BillToResellerID],
                    [$Table].[OrderID] as [OrderID],
                    [$Table].[DeliveryMethodID] as [DeliveryMethodID],
                    [$Table].[ContactPersonID] as [ContactPersonID],
                    [$Table].[AccountsPersonID] as [AccountsPersonID],
                    [$Table].[SalespersonPersonID] as [SalespersonPersonID],
                    [$Table].[PackedByPersonID] as [PackedByPersonID],
                    [$Table].[InvoiceDate] as [InvoiceDate],
                    [$Table].[CustomerPurchaseOrderNumber] as [CustomerPurchaseOrderNumber],
                    [$Table].[IsCreditNote] as [IsCreditNote],
                    [$Table].[CreditNoteReason] as [CreditNoteReason],
                    [$Table].[Comments] as [Comments],
                    [$Table].[DeliveryInstructions] as [DeliveryInstructions],
```

```
                [$Table].[InternalComments] as [InternalComments],
                [$Table].[TotalDryItems] as [TotalDryItems],
                [$Table].[TotalChillerItems] as [TotalChillerItems],
                [$Table].[DeliveryRun] as [DeliveryRun],
                [$Table].[RunPosition] as [RunPosition],
                [$Table].[ReturnedDeliveryData] as [ReturnedDeliveryData],
                [$Table].[ConfirmedDeliveryTime] as [ConfirmedDeliveryTime],
                [$Table].[ConfirmedReceivedBy] as [ConfirmedReceivedBy],
                [$Table].[LastEditedBy] as [LastEditedBy],
                [$Table].[LastEditedWhen] as [LastEditedWhen]
            from [lh_FAIAD].[dbo].[Invoices] as [$Table]
            union all select [$Table].[InvoiceID] as [InvoiceID],
                [$Table].[CustomerID] as [CustomerID],
                [$Table].[BillToResellerID] as [BillToResellerID],
                [$Table].[OrderID] as [OrderID],
                [$Table].[DeliveryMethodID] as [DeliveryMethodID],
                [$Table].[ContactPersonID] as [ContactPersonID],
                [$Table].[AccountsPersonID] as [AccountsPersonID],
                [$Table].[SalespersonPersonID] as [SalespersonPersonID],
                [$Table].[PackedByPersonID] as [PackedByPersonID],
                [$Table].[InvoiceDate] as [InvoiceDate],
                [$Table].[CustomerPurchaseOrderNumber] as [CustomerPurchaseOrderNumber],
                [$Table].[IsCreditNote] as [IsCreditNote],
                [$Table].[CreditNoteReason] as [CreditNoteReason],
                [$Table].[Comments] as [Comments],
                [$Table].[DeliveryInstructions] as [DeliveryInstructions],
                [$Table].[InternalComments] as [InternalComments],
                [$Table].[TotalDryItems] as [TotalDryItems],
                [$Table].[TotalChillerItems] as [TotalChillerItems],
                [$Table].[DeliveryRun] as [DeliveryRun],
                [$Table].[RunPosition] as [RunPosition],
                [$Table].[ReturnedDeliveryData] as [ReturnedDeliveryData],
                [$Table].[ConfirmedDeliveryTime] as [ConfirmedDeliveryTime],
                [$Table].[ConfirmedReceivedBy] as [ConfirmedReceivedBy],
                [$Table].[LastEditedBy] as [LastEditedBy],
                [$Table].[LastEditedWhen] as [LastEditedWhen]
            from [lh_FAIAD].[dbo].[InvoicesMay] as [$Table]
          ) as [_]
        ) as [$Inner] on ([$Outer].[InvoiceID] = [$Inner].[InvoiceID2] or [$Outer].[InvoiceID] is null and
[$Inner].[InvoiceID2] is null)
    ) as [_]
) as [$Outer]
where exists
(
    select 1
    from
    (
        select [ResellerID]
        from [lh_FAIAD].[dbo].[Reseller] as [$Table]
    ) as [$Inner]
    where [$Outer].[CustomerID] = [$Inner].[ResellerID] or [$Outer].[CustomerID] is null and
[$Inner].[ResellerID] is null
)
```
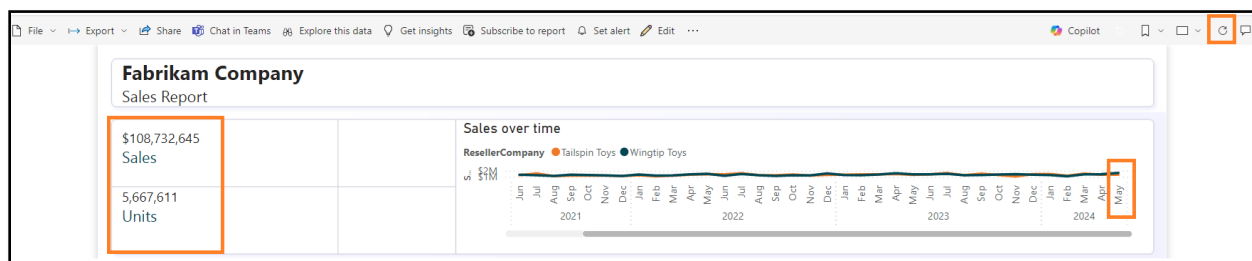
)

19. From the visual query menu, select **Run** to execute the code.

Once the code is executed, we have updated Sales table to include May 2024 data.
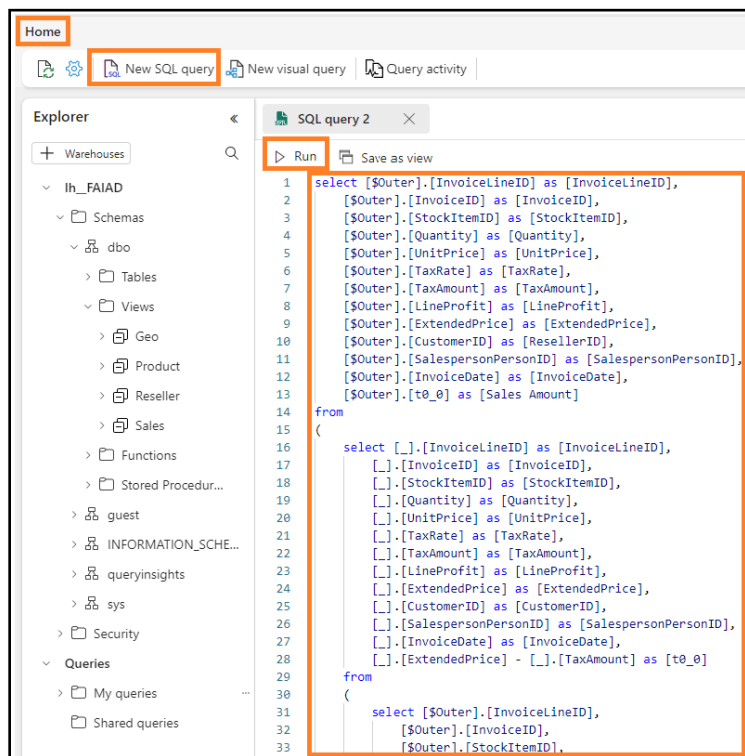
20. Select **rpt_Sales_Report** from the left menu bar to navigate back to the report**.**
21. From the top menu select **Refresh**. Notice now in the Line chart there is data for May 2024. Also, notice the Sales amount and Units has increased.

We do not have to refresh the data model and report when data changes. This is the advantage of Direct Lake and Direct query.

Let's revisit the challenges that are listed in the problem statement:

- **You need to refresh your dataset at least three times a day to accommodate the different update times for the different data sources.**
  We solved this using Direct Lake. Each individual Dataflow is refreshed on its schedule. Dataset and report do not have to be refreshed.

- **Your refreshes take a long time as you need to do a full refresh every time to capture any updates that happened to the source systems.**
  Again, we solved this using Direct Lake. Each individual Dataflow is refreshed on its schedule. Dataset and report do not have to be refreshed, so we do not have to worry about full refresh.
- **Any errors in any of the data sources that you are pulling from will result in your dataset refresh breaking. A lot of times the employee file doesn't upload on time resulting in your dataset refresh breaking.**
  Data Pipeline helps to solve this problem, by providing the ability to retry refresh on failure and at different intervals.
- **It takes a very long time to make any changes to your data model as Power Query takes a long time to refresh your previews, given the large data sizes and complex transformations.**
  We noticed Dataflows and Lakehouse are efficient and easy to make changes. Typically, preview in Dataflows and Lakehouse do not take long to load.
- **You need a Windows PC to use Power BI Desktop even though the corporate standard is Mac.**
  Microsoft Fabric is a SaaS offering. All we need is a browser to access the service. We do not have to install any software on our desktops.

Clean up Lab environment

Once you are ready to clean up the lab environment, follow the steps below.

1. Select **FAIAD_<username>** workspace from the left panel to navigate to the workspace home.
2. From the top menu, select **Workspace settings**.
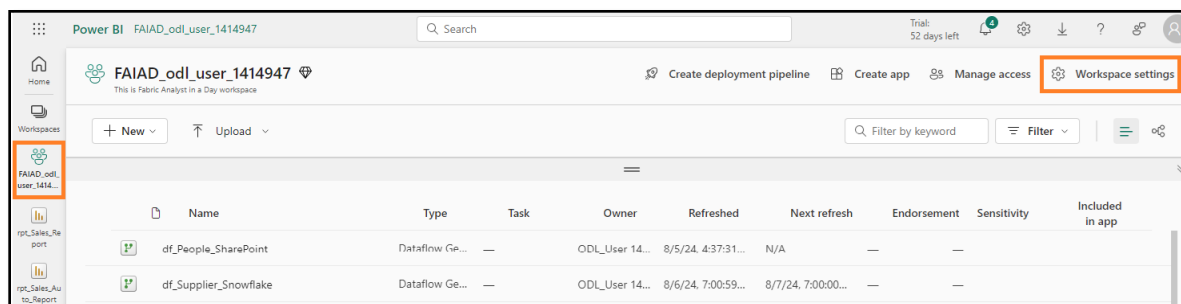
3. Workspace settings dialog opens. In **General** section, scroll down.
4. Select **Remove this workspace**.
5. Delete workspace dialog opens. Select **Delete**.

This will delete the workspace and all the items that were contained in the workspace.
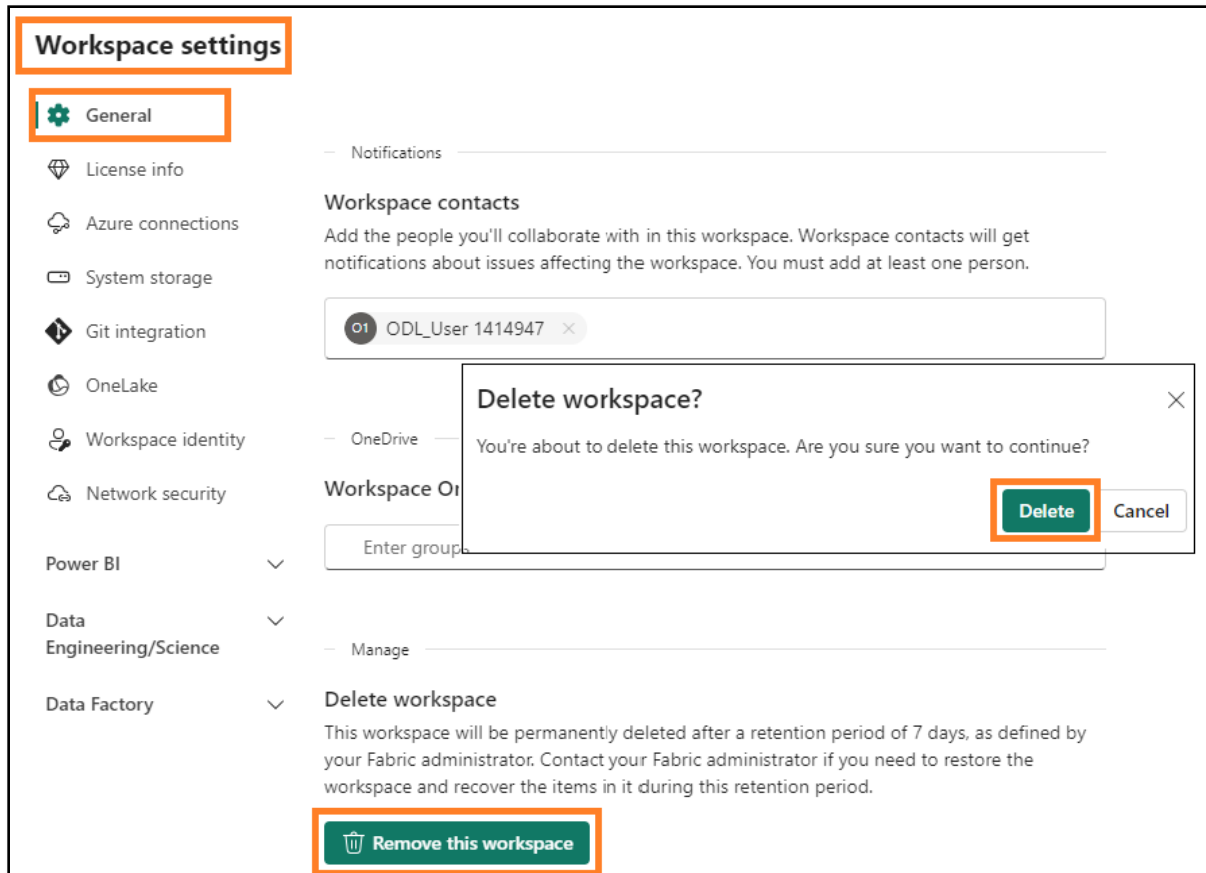
References

Fabric Analyst in a Day (FAIAD) introduces you to some of the key functions available in Microsoft Fabric. In the menu of the service, the Help (?) section has links to some great resources.
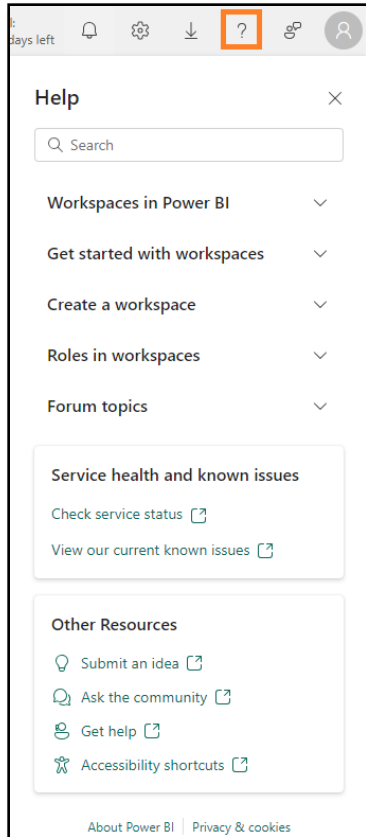
Here are a few more resources that will help you with your next steps with Microsoft Fabric.

- See blog post to read the full [Microsoft Fabric GA announcement](#)
- Explore Fabric through the [Guided Tour](#)
- Sign up for the [Microsoft Fabric free trial](#)
- Visit the [Microsoft Fabric website](#)
- Learn new skills by exploring the [Fabric Learning modules](#)
- Explore the [Fabric technical documentation](#)
- Read the [free e-book on getting started with Fabric](#)
- Join the [Fabric community](#) to post your questions, share your feedback, and learn from others

Read the more in-depth Fabric experience announcement blogs:

- [Data Factory experience in Fabric blog](#)
- [Synapse Data Engineering experience in Fabric blog](#)
- [Synapse Data Science experience in Fabric blog](#)
- [Synapse Data Warehousing experience in Fabric blog](#)
- [Synapse Real-Time Analytics experience in Fabric blog](#)
- [Power BI announcement blog](#)
- [Data Activator experience in Fabric blog](#)
- [Administration and governance in Fabric blog](#)
- [OneLake in Fabric blog](#)

- [Dataverse and Microsoft Fabric integration blog](#)

DISCLAIMER

This demo/lab contains only a portion of new features and enhancements in Microsoft Power BI. Some of the features might change in future releases of the product. In this demo/lab, you will learn about some, but not all, new features.