Lab 03: Implementing Data Pipelines with Fabric Data Factory

Introduction

In this lab, you will ingest parquet containing sales data directly to the Managed Lakehouse table, using a Data Pipeline with Copy activity, to append this dataset to the existing Sales table.

Objectives

After completing this lab, you will be better able to:

1. Create a data pipeline

2. Execute and monitor the pipeline

3. Create a basic Semantic Model

4. Auto-create a Power BI report to visualize data

Estimated time to complete this lab

60 minutes

Contents

Lab Prerequisites

- Workspace: Fabric, Power Premium or Fabric trial
- Individual license: Power Pro or Premium Per User account

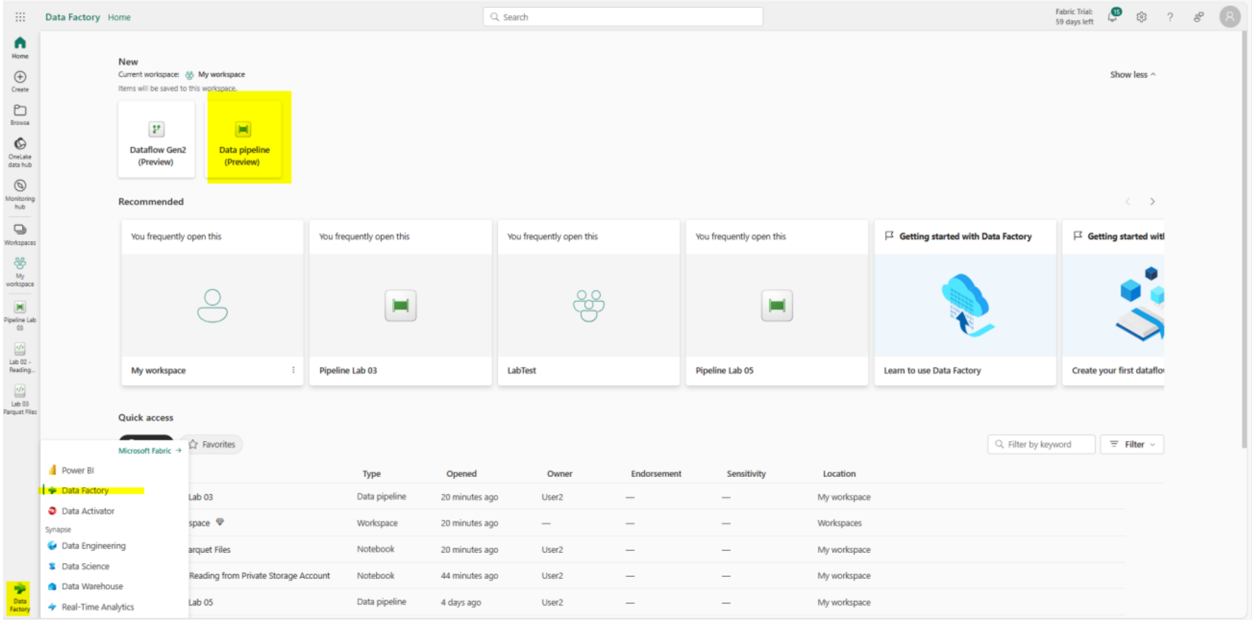
Information provided by your training provider

- Trial tenant (if applicable): login & password, workspace to use for the lab.
- Azure Data Lake Gen2 (containing data sources): account name & shared access signature.

Task 1: Create a Data pipeline

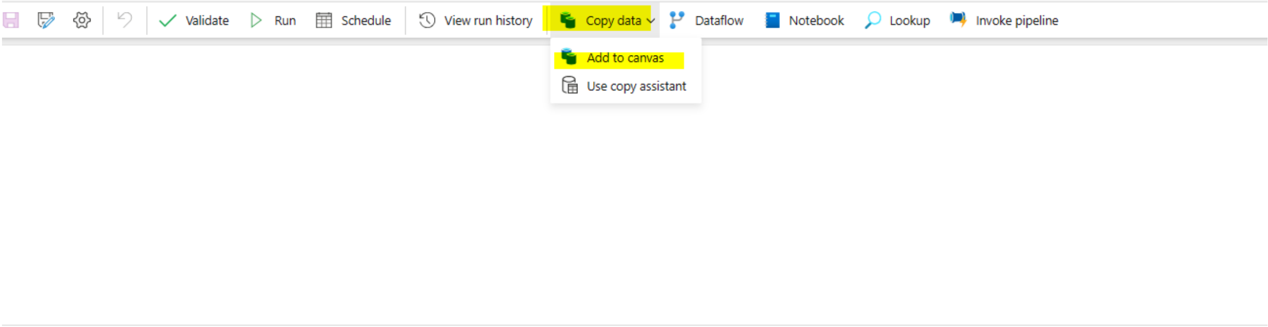
In this task, you will create the Copy Activity that copies the parquet files from a private storage account to the Sales table from the Managed Lakehouse.

- On the MS Fabric menu, navigate to Data Factory and click on the Data pipeline. Provide the name Pipeline Lab 03



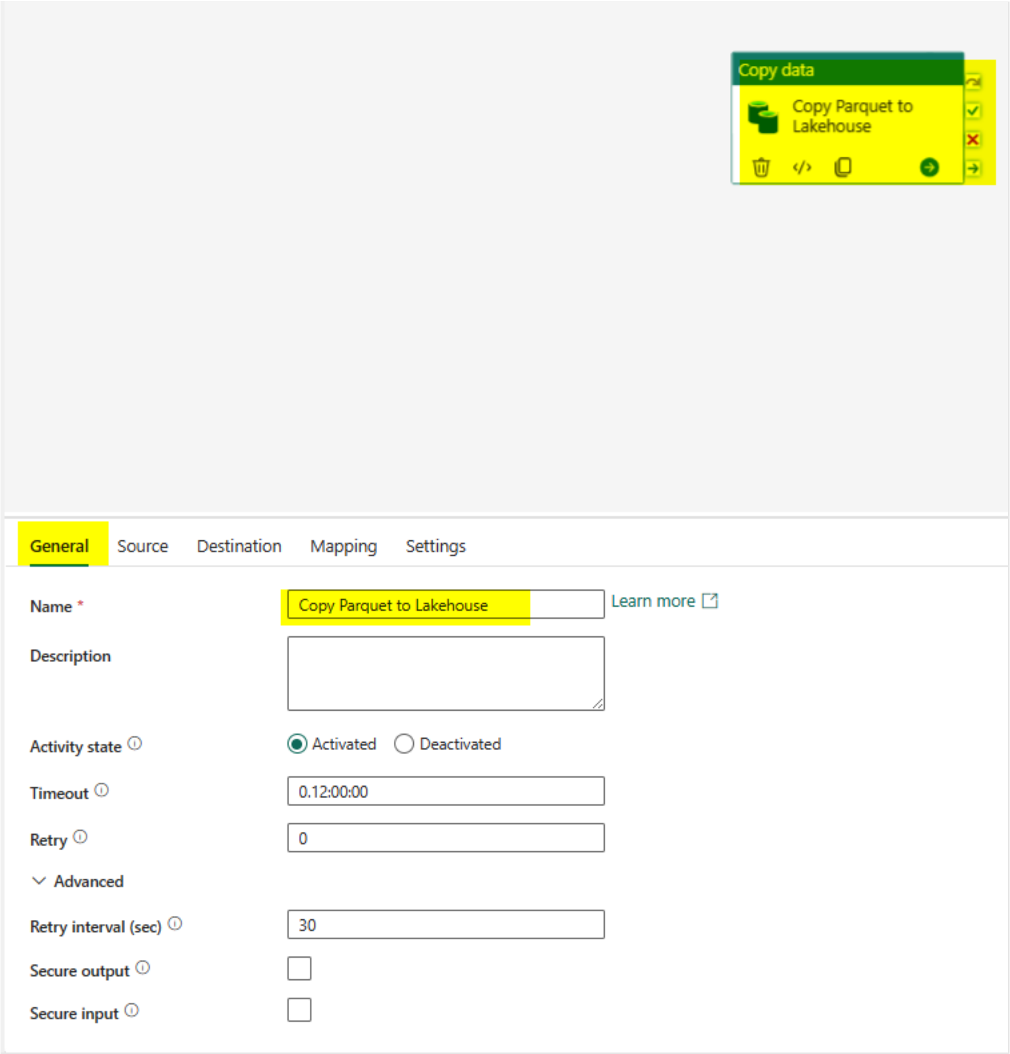
	Type	Opened	Owner	Endorsement	Sensitivity	Location
Lab 03	Data pipeline	20 minutes ago	User2	—	—	My workspace
space	Workspace	20 minutes ago	—	—	—	Workspaces
arquet Files	Notebook	20 minutes ago	User2	—	—	My workspace
Reading from Private Storage Account	Notebook	44 minutes ago	User2	—	—	My workspace
Lab 05	Data pipeline	4 days ago	User2	—	—	My workspace

- From the top menu, click on the Copy data and select Add to canvas:



- Click on the Copy data that you just added to the Canvas:

Go to the **General** Tab and under the Name type: Copy Parquet to Lakehouse. The other settings under this tab will remain unchanged.



Name * Copy Parquet to Lakehouse

Description

Activity state ● Activated ○ Deactivated

Timeout 0.12:00:00

Retry 0

Advanced

Retry interval (sec) 30

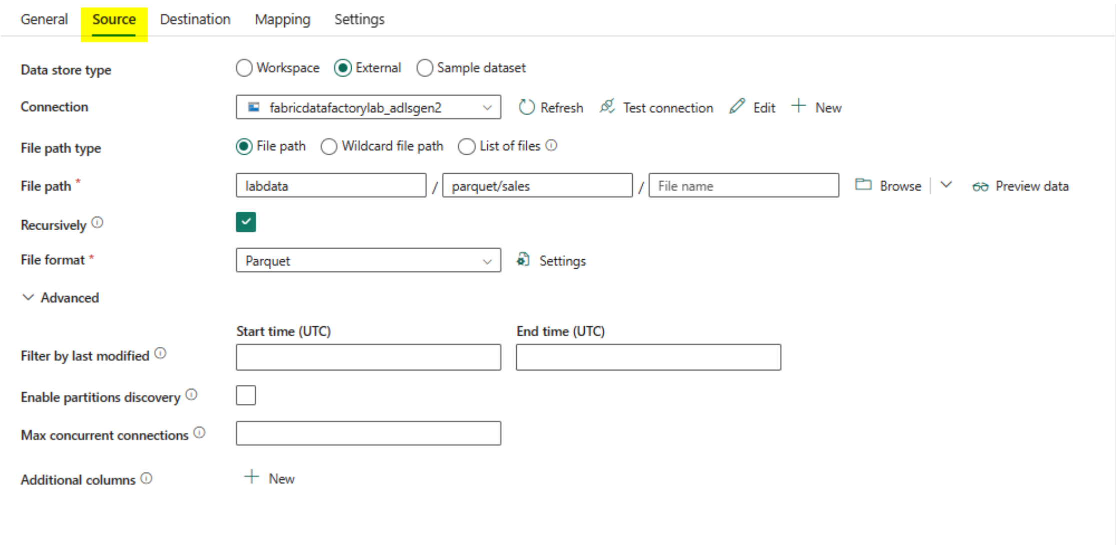
Secure output □

Secure input □

- Click on Source Tab and select the following:

- Data store type: External
- Connection: Choose the name you provided in the first lab when you connected to the Private Storage account and test the connection to be successful
- File path type: File path
- File path: click on browse and navigate to the container labdata/parquet/sales
- **Recursively**: checked (this will ensure that all the files from the subfolders labdata/parquet/sales will be read)
- File format: Parquet
- All the other settings from this tab will remain unchanged.

Make sure that your Source tab looks like this at the end:



General | **Source** | Destination | Mapping | Settings

Data store type: ○ Workspace ● External ○ Sample dataset

Connection: fabricdatafactorylab_adlsgen2 — Refresh, Test connection, Edit, + New

File path type: ● File path ○ Wildcard file path ○ List of files ⓘ

File path *: labdata / parquet/sales / File name — Browse, Preview data

Recursively ⓘ: ☑

File format *: Parquet — Settings

Advanced

Filter by last modified ⓘ: Start time (UTC) | End time (UTC)

Enable partitions discovery ⓘ: □

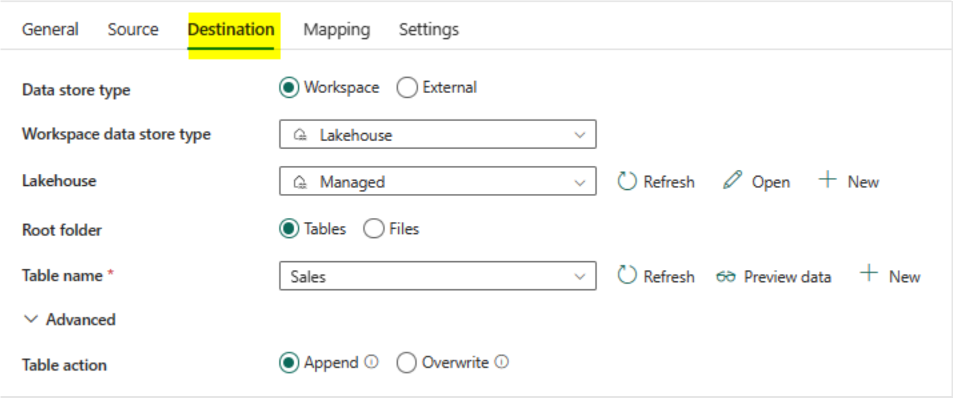
Max concurrent connections ⓘ:

Additional columns ⓘ: + New

- Click on the Destination tab:

- Data store type: Workspace
- Workspace data store type: Lakehouse
- Lakehouse: Managed(this should be the name you provided on the first lab)
- Root folder: Tables
- Table name: Sales
- Expand Advanced
  - – Table action: Append

At the end the Destination tab should look like this:



General | Source | **Destination** | Mapping | Settings

Data store type: ● Workspace ○ External

Workspace data store type: Lakehouse

Lakehouse: Managed — Refresh, Open, + New

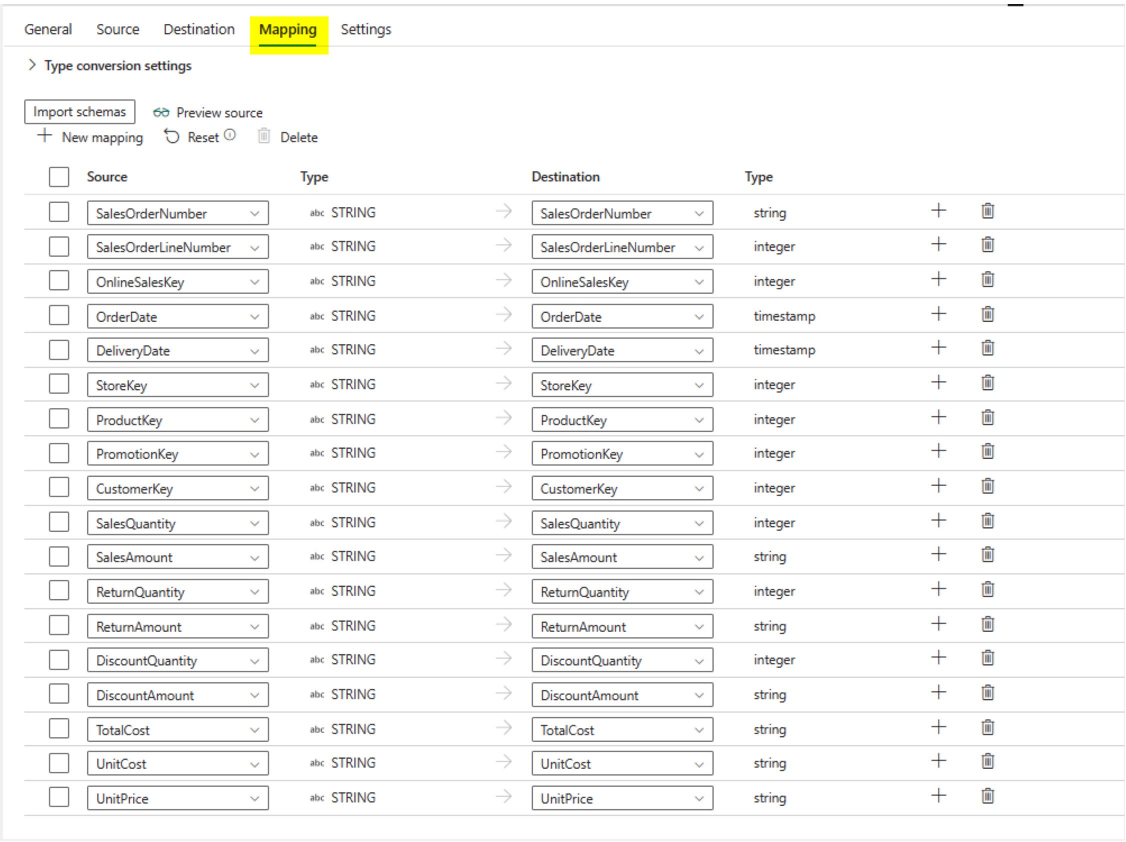
Root folder: ● Tables ○ Files

Table name *: Sales — Refresh, Preview data, + New

Advanced

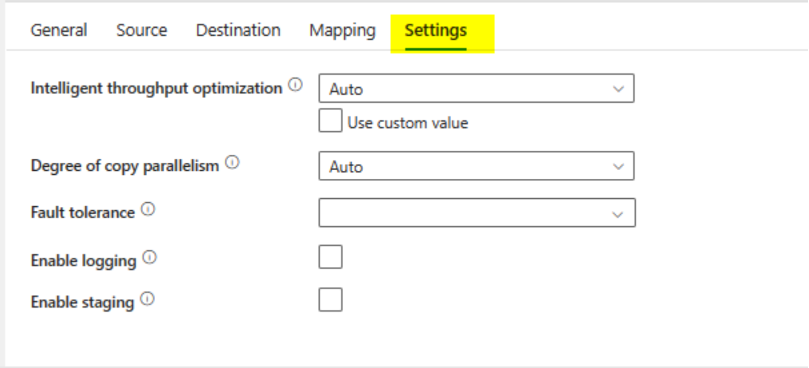
Table action: ● Append ⓘ ○ Overwrite ⓘ

- On the Mapping tab, the mapping should be done automatically. If not, click on the Import schemas. At the end the Mapping tab should look like this:



Source	Type	Destination	Type
SalesOrderNumber	STRING	SalesOrderNumber	string
SalesOrderLineNumber	STRING	SalesOrderLineNumber	integer
OnlineSalesKey	STRING	OnlineSalesKey	integer
OrderDate	STRING	OrderDate	timestamp
DeliveryDate	STRING	DeliveryDate	timestamp
StoreKey	STRING	StoreKey	integer
ProductKey	STRING	ProductKey	integer
PromotionKey	STRING	PromotionKey	integer
CustomerKey	STRING	CustomerKey	integer
SalesQuantity	STRING	SalesQuantity	integer
SalesAmount	STRING	SalesAmount	string
ReturnQuantity	STRING	ReturnQuantity	integer
ReturnAmount	STRING	ReturnAmount	string
DiscountQuantity	STRING	DiscountQuantity	integer
DiscountAmount	STRING	DiscountAmount	string
TotalCost	STRING	TotalCost	string
UnitCost	STRING	UnitCost	string
UnitPrice	STRING	UnitPrice	string

- The Settings tab remains unchanged:



Intelligent throughput optimization: Auto

□ Use custom value

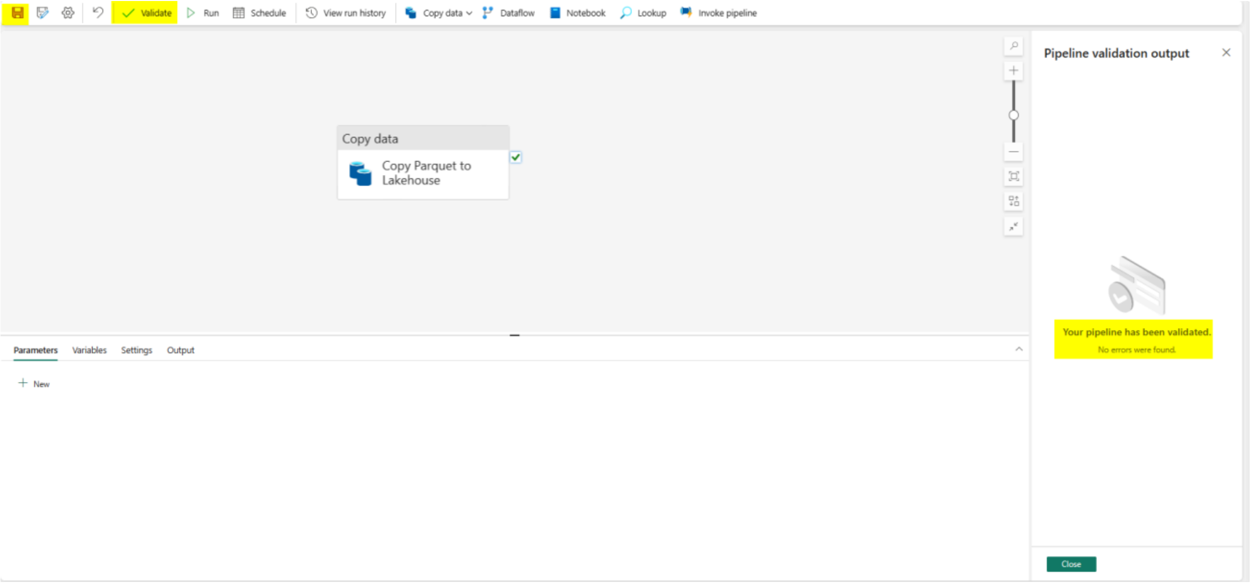
Degree of copy parallelism: Auto

Fault tolerance:

Enable logging □

Enable staging □

- Once you finish building the Pipeline, from the top tab, click on the Validate to make sure that there are no errors and then finally on the Save button.



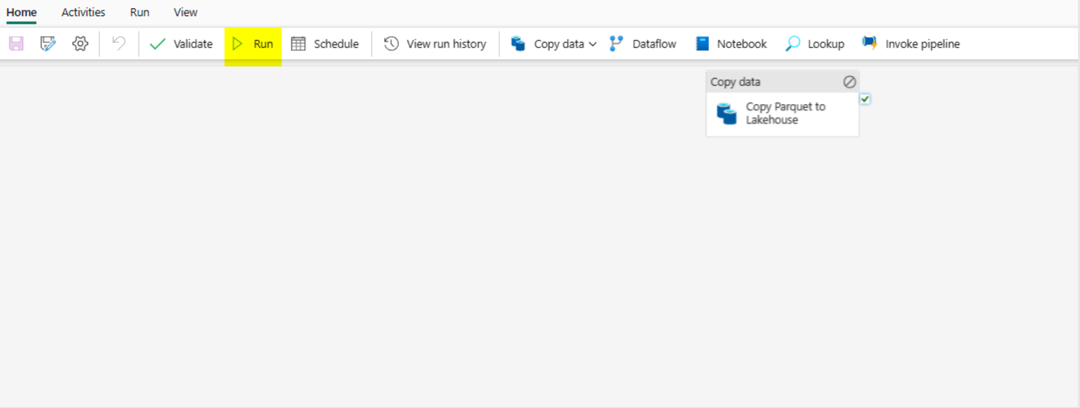
Your pipeline has been validated.

No errors were found.

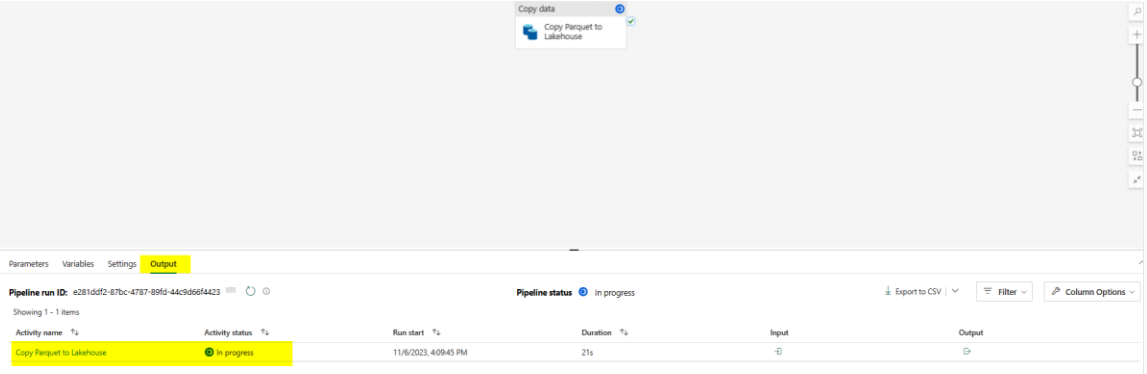
Task 2: Execute and monitor the pipeline

In this task, you will start by trigger the pipeline

- From the top menu click on the Run button.



Then at the button of the page click on the Output and then on the Pipeline:



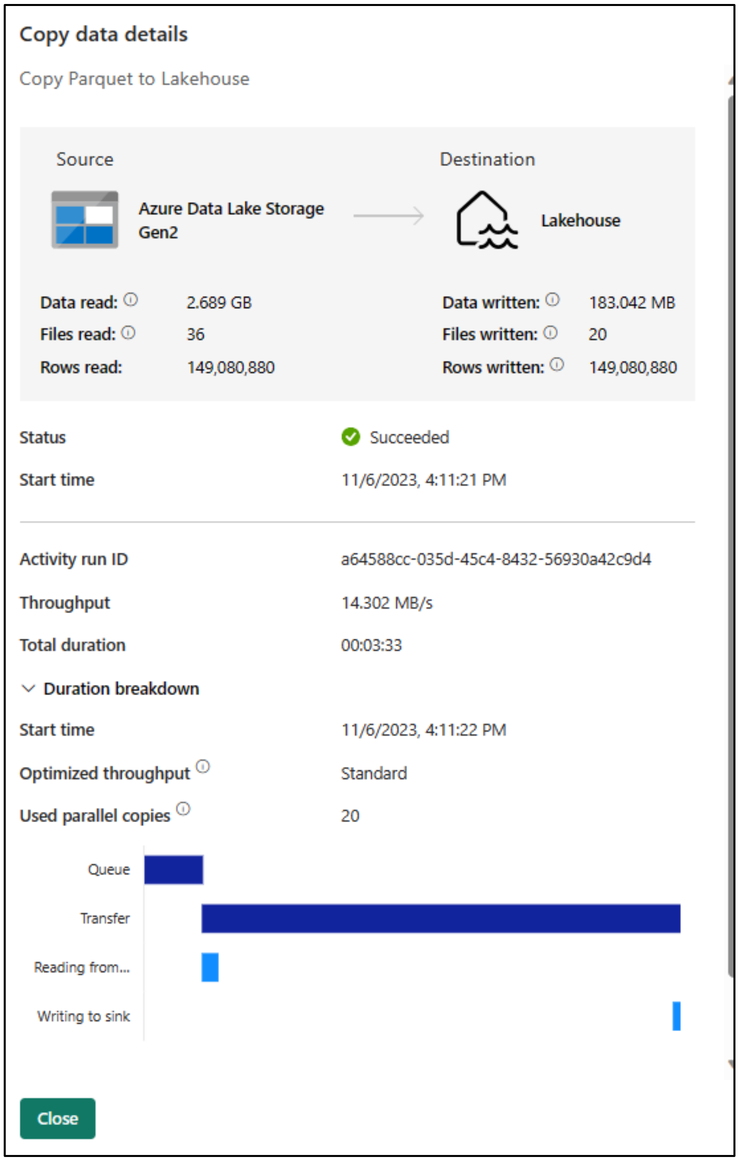
Pipeline run ID: e281ddf2-87bc-4787-89fd-44c9d66f4423

Pipeline status In progress

Showing 1 - 1 items

Activity name	Activity status	Run start	Duration	Input	Output
Copy Parquet to Lakehouse	In progress	11/6/2023, 4:09:45 PM	21s		

You can explore the copy data details to understand how much data you have transferred to the Lakehouse table.



Copy data details

Copy Parquet to Lakehouse

Source		Destination	
Azure Data Lake Storage Gen2		Lakehouse	
Data read:	2.689 GB	Data written:	183.042 MB
Files read:	36	Files written:	20
Rows read:	149,080,880	Rows written:	149,080,880

Status	Succeeded
Start time	11/6/2023, 4:11:21 PM
Activity run ID	a64588cc-035d-45c4-8432-56930a42c9d4
Throughput	14.302 MB/s
Total duration	00:03:33

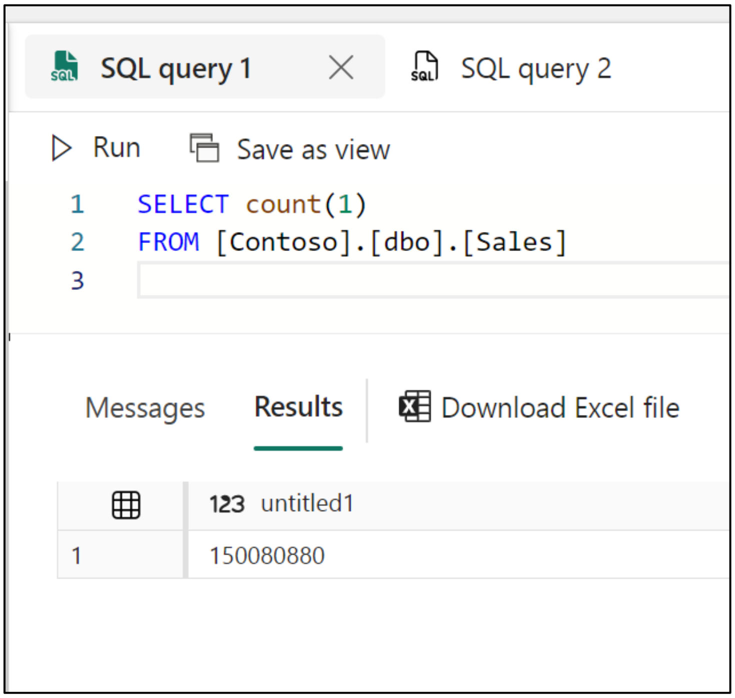
Duration breakdown

Start time	11/6/2023, 4:11:22 PM
Optimized throughput	Standard
Used parallel copies	20

As it can be observed, at the time we read the data from ADLS Gen 2 we had 2.689GB with 36 files, however, upon migration to the Lakehouse, the dataset transformed into a more compact form, occupying 183.042MB distributed across 20 files

This exercise illustrates the efficiency of transforming data from ADLS Gen 2 into a Delta table within the Lakehouse environment. By leveraging Delta tables, data compression is improved, and smaller files are consolidated into larger ones, optimizing data storage and access.

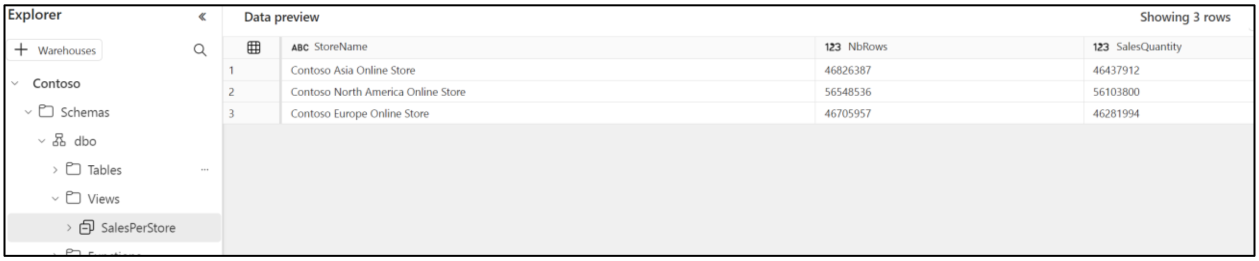
- Come back to the SQL Analytics Endpoint of your Lakehouse, and run this basic SQL query to count the number of rows.
- You can realize that the Sales now contains more than 150M of rows.



```
SELECT count(1)
FROM [Contoso].[dbo].[Sales]
```

	untitled1
1	150080880

- You can also use the previously created SQL View to control sales per store.

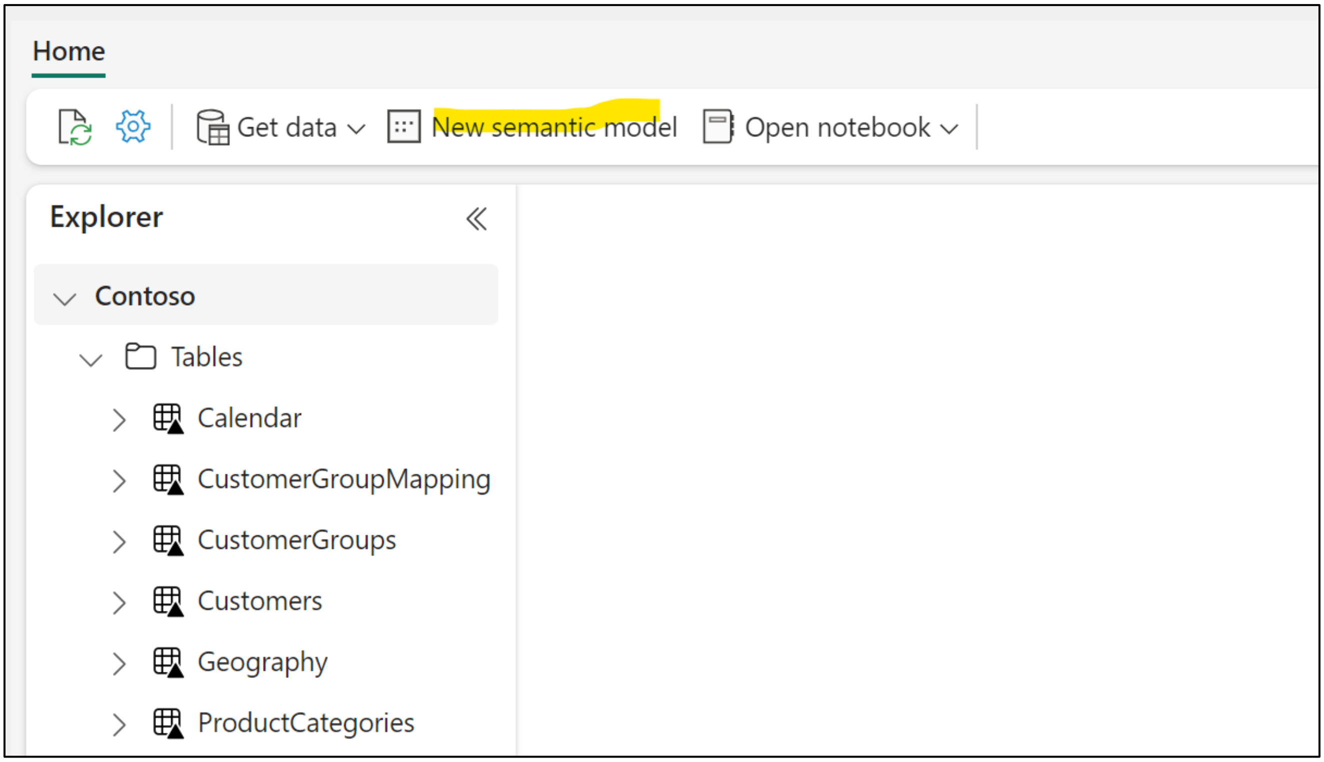


	StoreName	NbRows	SalesQuantity
1	Contoso Asia Online Store	46826387	46437912
2	Contoso North America Online Store	56548536	56103800
3	Contoso Europe Online Store	46705957	46281994

Task 3: Create the Semantic Model

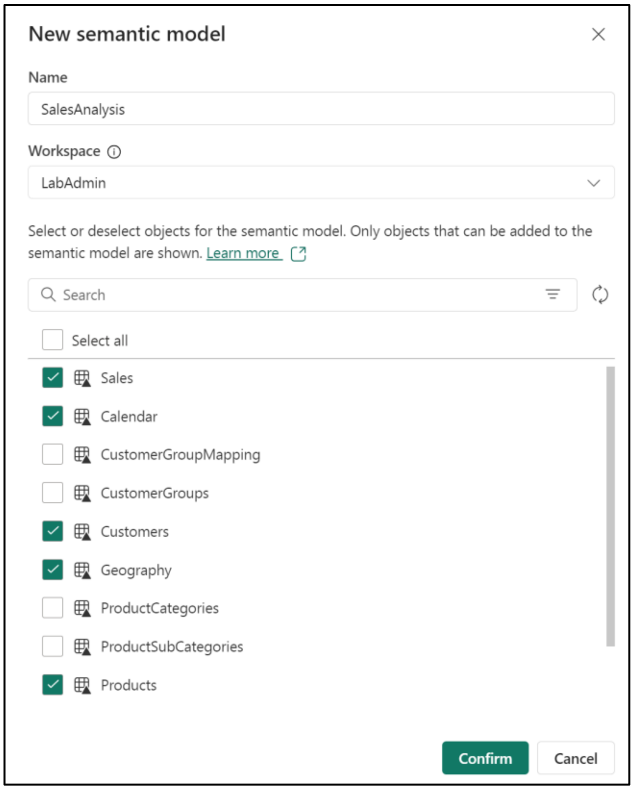
In this task, you will create a simple Semantic Model from the Lakehouse, based on the Direct Lakehouse Mode.

- From the Lab Workspace, select the Contoso Lakehouse
- In the Lakehouse,viewer, click on the New semantic model button



- On the New semantic model UI, type SalesAnalysis as semantic model name, and select the following tables :
  - Sales
  - Calendar
  - Customers
  - Geography
  - Products
  - Stores
  - Promotions

- Click on Submit to initiate the Dataset creation



New semantic model

Name

SalesAnalysis

Workspace ⓘ

LabAdmin

Select or deselect objects for the semantic model. Only objects that can be added to the semantic model are shown. Learn more

Search

☐ Select all

☑ Sales

☑ Calendar

☐ CustomerGroupMapping

☐ CustomerGroups

☑ Customers

☑ Geography

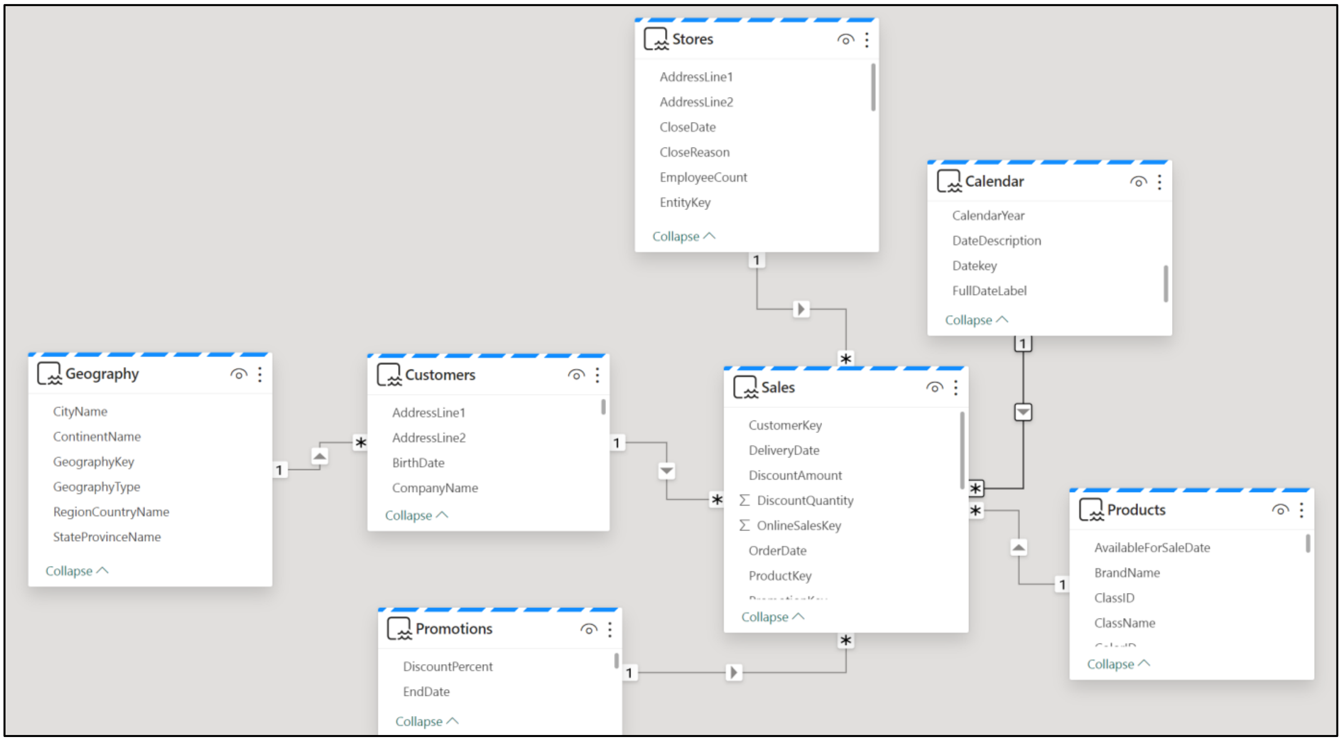
☐ ProductCategories

☐ ProductSubCategories

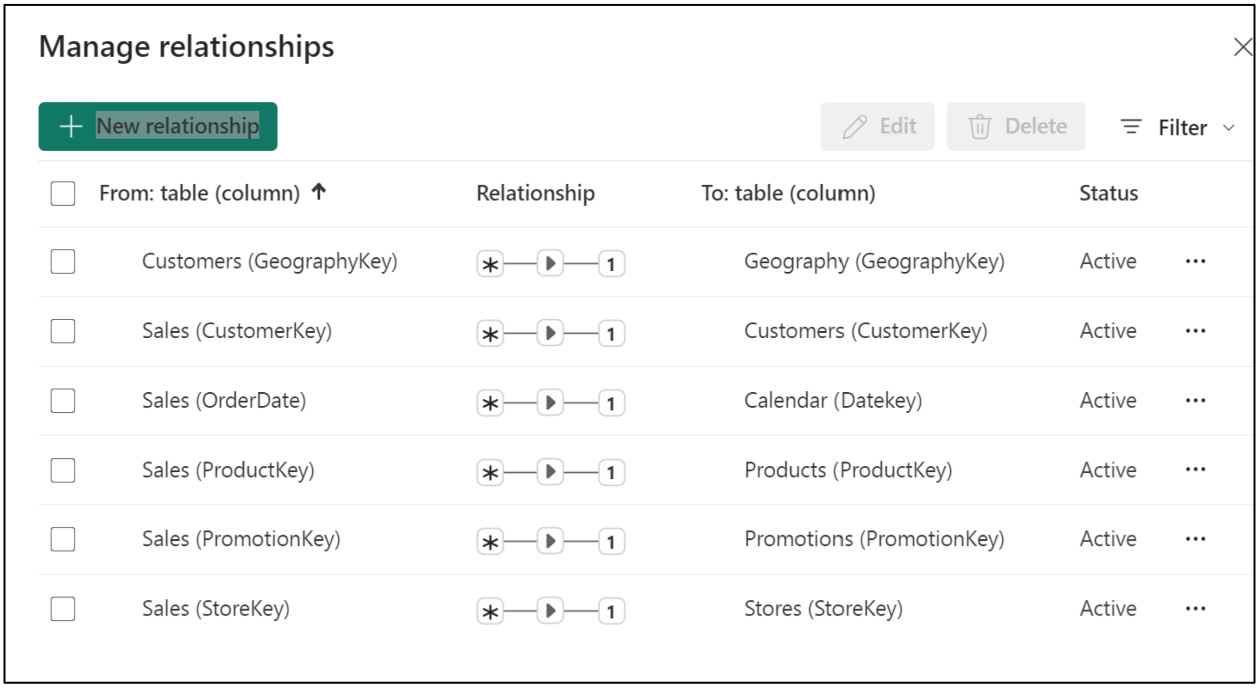
☑ Products

Confirm Cancel

- The objective is now to design the semantic model with the following relationships :



- Click on the Manage relationships button to create validate the requiered relationships between tables.



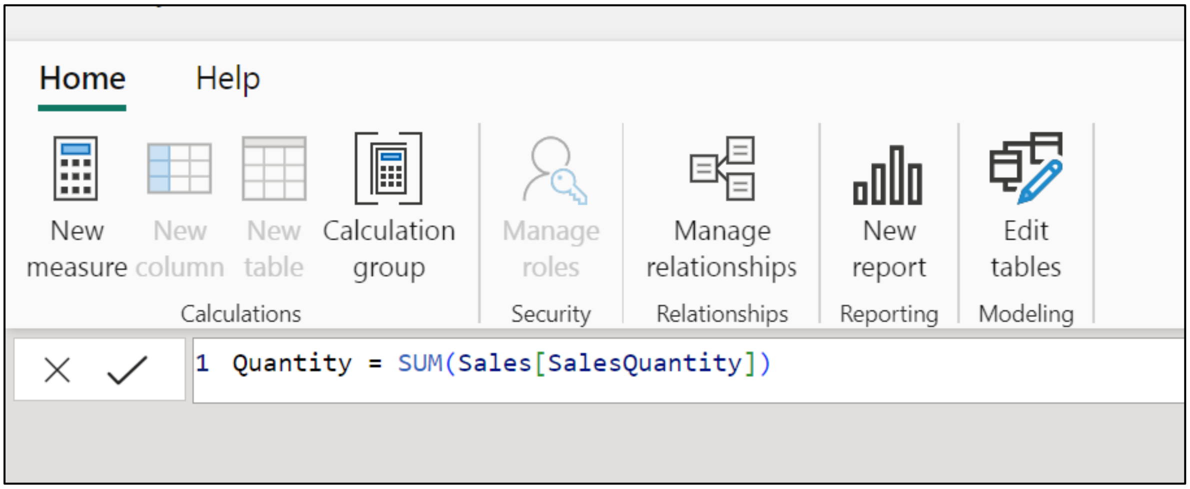
Manage relationships

□	From: table (column) ↑	Relationship	To: table (column)	Status	
□	Customers (GeographyKey)	*—▶—1	Geography (GeographyKey)	Active	···
□	Sales (CustomerKey)	*—▶—1	Customers (CustomerKey)	Active	···
□	Sales (OrderDate)	*—▶—1	Calendar (Datekey)	Active	···
□	Sales (ProductKey)	*—▶—1	Products (ProductKey)	Active	···
□	Sales (PromotionKey)	*—▶—1	Promotions (PromotionKey)	Active	···
□	Sales (StoreKey)	*—▶—1	Stores (StoreKey)	Active	···

- You will now implement 2 very simple Measures (business calculations)

- Create the measures Quantity using the following DAX definition:

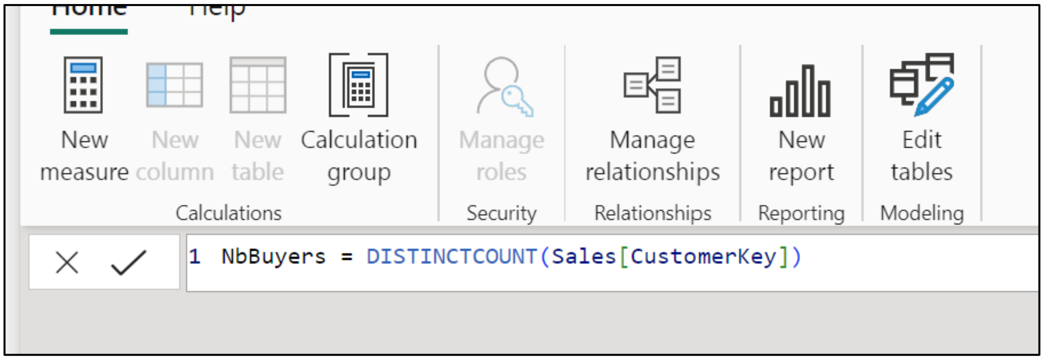
```
Quantity=SUM(Sales[SalesQuantity])
```



```
Quantity = SUM(Sales[SalesQuantity])
```

- Create the measures NbBuyers using the following DAX definition:

```
NbBuyers=DISTINCTCOUNT(Sales[CustomerKey])
```



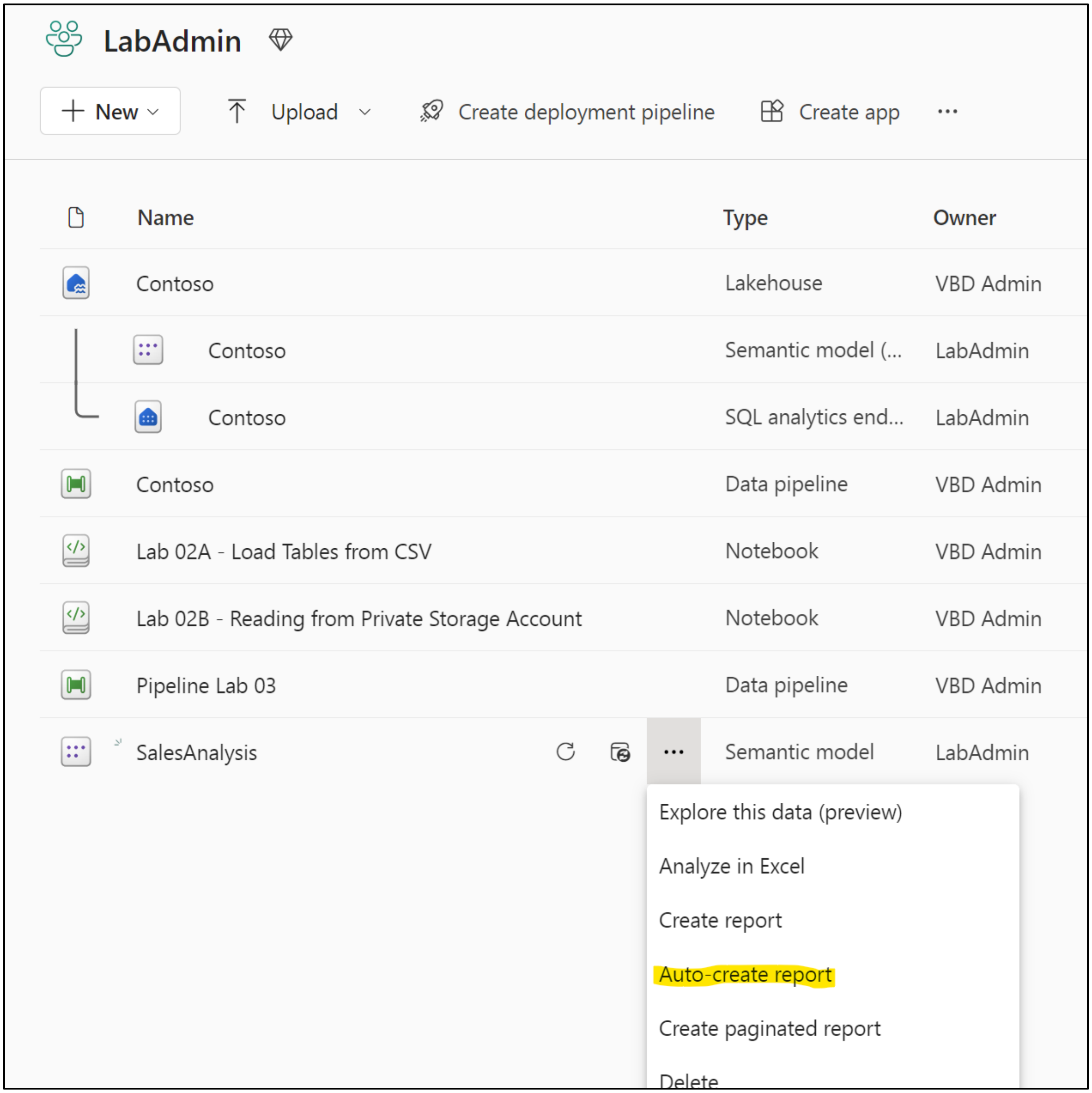
```
1  NbBuyers = DISTINCTCOUNT(Sales[CustomerKey])
```

The semantic model is now ready to support report visalizations.

Task 4: Create the Report

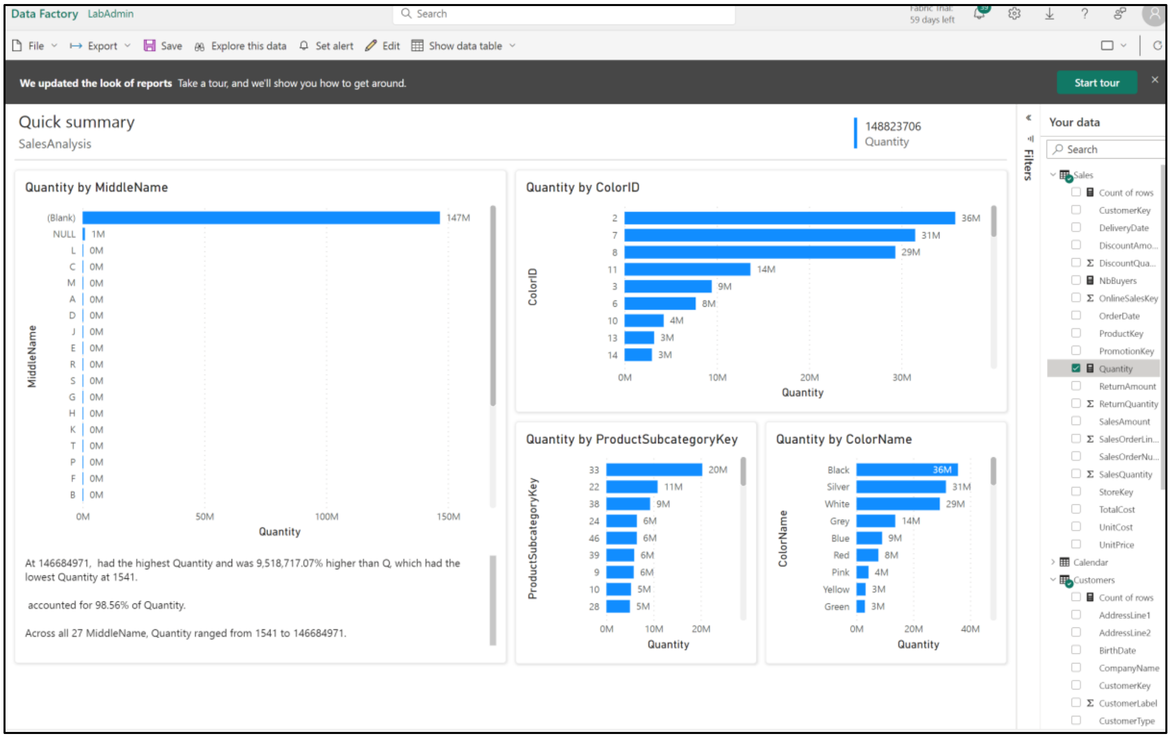
In this task, you will create a simple report to visualize data from the dataset using the Auto-create report function, then you will customize it.

- From the Lab workspace, select the SalesAnalysis semantic mode, click on the dotes and select the option Auto-create report



Name	Type	Owner
Contoso	Lakehouse	VBD Admin
Contoso	Semantic model (...	LabAdmin
Contoso	SQL analytics end...	LabAdmin
Contoso	Data pipeline	VBD Admin
Lab 02A - Load Tables from CSV	Notebook	VBD Admin
Lab 02B - Reading from Private Storage Account	Notebook	VBD Admin
Pipeline Lab 03	Data pipeline	VBD Admin
SalesAnalysis	Semantic model	LabAdmin

- After few seconds, a basic report is created



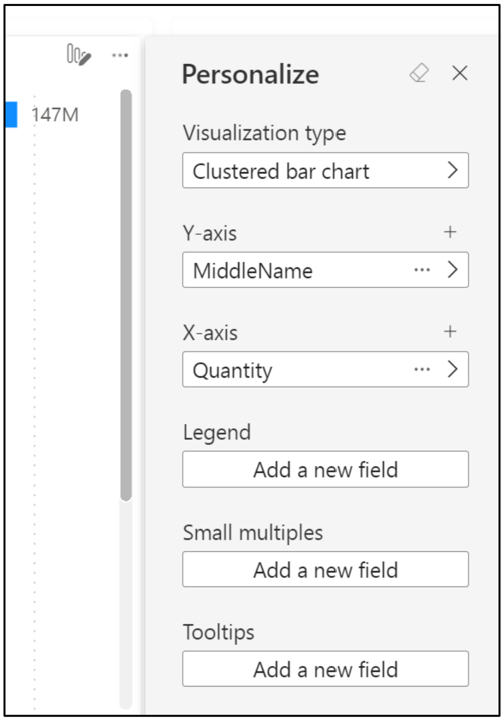
We updated the look of reports Take a tour, and we'll show you how to get around.

At 146684971, had the highest Quantity and was 9,518,717.07% higher than Q, which had the lowest Quantity at 1541.

accounted for 98.56% of Quantity.

Across all 27 MiddleName, Quantity ranged from 1541 to 146684971.

- Then you can personalize each visual by changing the Visualization type and the columns and measures displayed on the visual axis.



Visualization type

Clustered bar chart

Y-axis

MiddleName

X-axis

Quantity

Legend

Add a new field

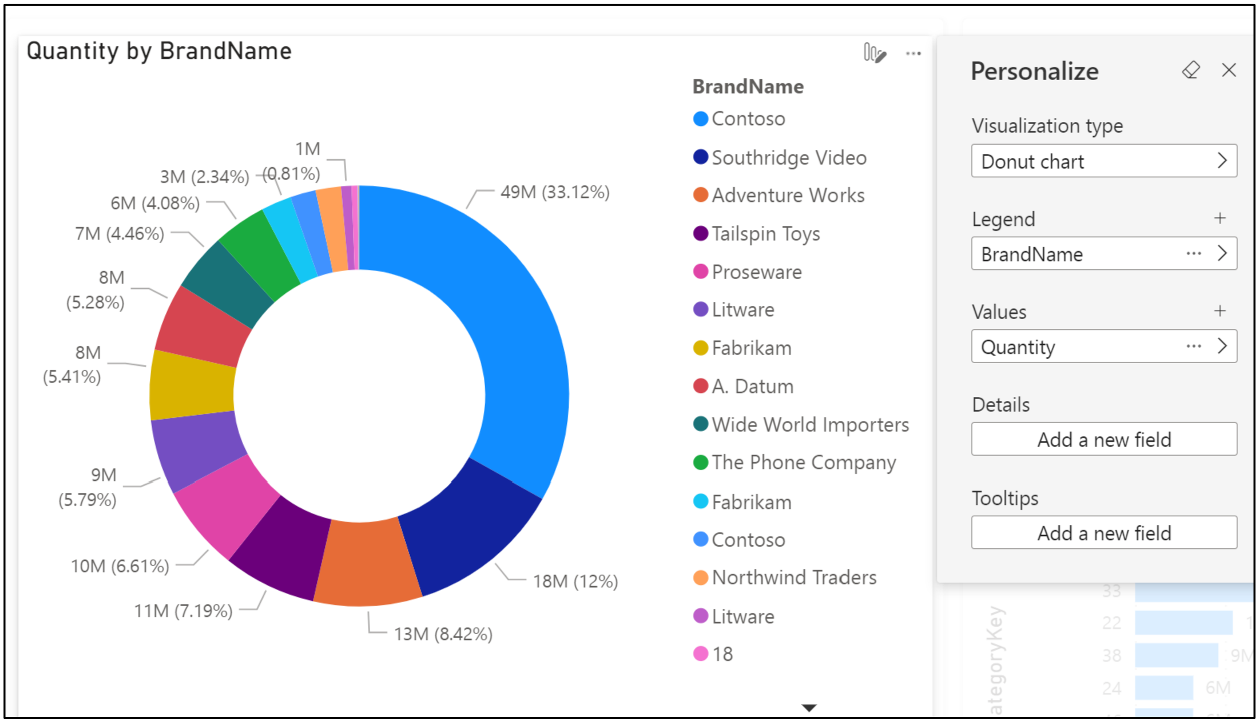
Small multiples

Add a new field

Tooltips

Add a new field

- The screenshot hereafter displays the result using a donut chart with Sales quantity per Product Brands names.



Personalize

Visualization type: Donut chart

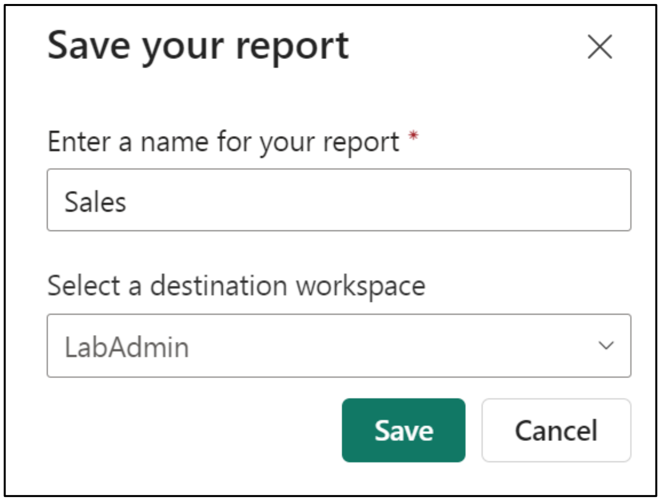
Legend: BrandName

Values: Quantity

Details: Add a new field

Tooltips: Add a new field

- Finally, save your report



Save your report

Enter a name for your report *: Sales

Select a destination workspace: LabAdmin