

1. Lakehouse

A Lakehouse is a data architecture platform for storing, managing, and analyzing structured and unstructured data in a single location. It is a flexible and scalable solution that allows organizations to handle large volumes of data using a variety of tools and frameworks to process and analyze that data.

1.1 Create Workspace

The screenshot shows the Microsoft Synapse Data Engineering workspace creation interface. The top navigation bar includes 'Synapse Data Engineering' and 'My workspace'. A search bar is at the top right. On the left, a sidebar lists 'Home', 'Create' (highlighted), 'Browse', 'OneLake data hub', 'Monitoring hub', and 'Workspaces' (highlighted). A 'My workspace' section shows three existing workspaces: 'My workspace' (selected), 'Fabric1', 'Fabric2', and 'Fabric3'. Below this, there are several preview cards: 'Notebook (Preview)', 'Spark Job Definition (Preview)', 'Data pipeline (Preview)', 'Experiment (Preview)', and 'Notebook (Preview)'. At the bottom left, a green button says '+ New workspace' with a plus sign icon. A trial status 'Trial: 59 days left' is in the top right corner.

- Click **+ New workspace**

Create a workspace

Name *

Fabric4

Available

Description

Fabric4

Domain (preview) ⓘ

Assign to a domain (optional)



Learn more about workspace settings

Workspace image



Upload

Reset

Advanced ^

Contact list * ⓘ



htran (Owner)



Enter users and groups

License mode ⓘ

Pro

Select Pro to use basic Power BI features and collaborate on reports, dashboards, and scorecards. To access a Pro workspace, users need Pro per-user licenses. [Learn more](#)

Trial

Select the free trial per-user license to try all the new features and experiences in Microsoft Fabric for 60 days. A Microsoft Fabric trial license allows users to create Microsoft Fabric items and collaborate with others in a Microsoft Fabric trial capacity. Explore new capabilities in Power BI, Data Factory, Data Engineering, and Real-Time Analytics, among others. [Learn more](#)

Apply

Cancel

- After input workspace name
- Select **Trial**
- Click **Apply**

1.2 Adding Lakehouse

Fabric4

Home

Create

Upload

Create app

Manage access

Workspace settings

Browse

OneLake data hub

Monitoring hub

Workspaces

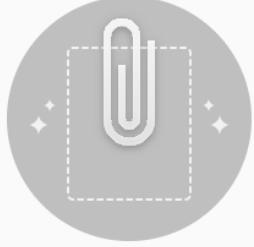
Fabric4

Microsoft Fabric →

- Power BI
- Data Factory
- Synapse
- Data Engineering**
- Data Science
- Data Warehouse
- Real-Time Analytics

There's nothing here yet

Add something new, or upload something to see them here.



- Navigate to **Data Engineering** icon > click  **Data Engineering**

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

Data Engineering

New

Current workspace: Fabric4

Items will be saved to this workspace.

Lakehouse (Preview) (Selected)

Notebook (Preview)

Spark Job Definition ...

Data pipeline (Preview)

Import notebook

Use a sample

Recommended: Fabric3: You frequently open this

You frequently open this

Fabric3

You frequently open this

Lakehouse3_01

You frequently open this

Fabric1

You frequently open this

My workspace

You frequently open this

Fabric2

Quick access

Recent

Favorites

| | Name | Type | Opened | Owner | Endorsement | Sensitivity | Location |
|---|----------------|-----------|----------------|---------|-------------|-------------|------------|
| 👤 | Fabric4 📄 | Workspace | now | — | — | — | Workspaces |
| 👤 | My workspace 📄 | Workspace | 14 minutes ago | — | — | — | Workspaces |
| 📊 | PowerBI | Report | a day ago | Fabric1 | — | — | Fabric1 |

- Click in Lakehouse icon

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

Data Engineering

New

Current workspace: Fabric4

Items will be saved to this workspace.

Lakehouse (Preview)

Notebook (Preview)

Spark Job Definition ...

Data pipeline (Preview)

Import notebook

Use a sample

Recommended

You frequently open this

Fabric3

You frequently open this

Lakehouse3_01

You frequently open this

Fabric1

You frequently open this

My workspace

New lakehouse

Name *

lakehouse

Create Cancel

Quick access

Recent

Favorites

| | Name | Type | Opened | Owner | Endorsement | Sensitivity | Location |
|---|----------------|-----------|----------------|---------|-------------|-------------|------------|
| 👤 | Fabric4 📄 | Workspace | now | — | — | — | Workspaces |
| 👤 | My workspace 📄 | Workspace | 14 minutes ago | — | — | — | Workspaces |
| 📊 | PowerBI | Report | a day ago | Fabric1 | — | — | Fabric1 |

- Input lakehouse name

1.3 Dataflow Gen2

Dataflow Gen2 is for loading data from many sources. It looks like Power query in Power BI

The screenshot shows the Power BI Dataflow Gen2 interface. On the left, there's a sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and Data Engineering. The main area has a search bar at the top. Below it, there's a navigation bar with 'lakehouse1' and 'No label'. A dropdown menu is open under 'Get data' with options: 'Upload files', 'New data pipeline', 'New Power BI dataset', and 'Open notebook'. The 'Explorer' section shows a tree view with 'lakehouse1' expanded, showing 'Tables' and 'Files'. A red arrow points to the 'New Dataflow Gen2' option in the dropdown. The central part of the screen has a heading 'Get data in your lakehouse' and four cards: 'New Dataflow Gen2' (with a gear icon), 'New data pipeline' (with a cylinder icon), 'Open notebook' (with a document icon), and 'New shortcut' (with a folder icon).

Bank query

- Using Power Query M to add Date dimension table

The screenshot shows the Power Query interface. The top navigation bar includes 'Dataflow 1', 'No label', and a search bar. The 'Home' tab is selected. The ribbon has sections for Power Query (Home, Transform, Add column, View, Help), Get data (Get data, Enter data, Manage connections), Options (Options, Manage parameters), Refresh (Refresh, Properties, Advanced editor, Manage), Add data destination (Add data destination), Query (Choose columns, Remove columns, Manage columns), Reduce rows (Keep rows, Remove rows, Filter rows), Sort (Sort), Transform (Split column, Group by, Replace values), and Merge queries (Merge queries, Append queries, Combine files, Combine). A dropdown menu under 'Get data' is open, showing 'Blank query' highlighted with a red arrow. Below the ribbon, there are four cards: 'Import from Excel' (Excel icon), 'Import from SQL Server' (cylinder icon), 'Import from a Text/CSV file' (document icon), and 'Import from dataflows' (tree icon). At the bottom, there are links: 'Get data from another source →' and 'Import from a Power Query template'.

- Select Blank query

```

1 let
2     StartDate = #date(2012,01,01),
3     EndDate = Date.EndOfYear(Date.AddYears(Date.From(DateTime.FixedLocalNow()),1)),
4     Source = List.Dates(StartDate,
5         Duration.Days(EndDate-StartDate) +1,
6         #duration(1,0,0,0)),
7     #"Converted to Table" = Table.FromList(Source, Splitter.SplitByNothing(), null, null, ExtraValues.Error),
8     #"Renamed Columns" = Table.RenameColumns(#"Converted to Table",{"Column1", "Date"}),
9     #"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"Date", type date}}),
10    #"Added Custom" = Table.AddColumn(#"Changed Type", "StartOfMonth", each Date.StartOfMonth([Date])),
11    #"Added Custom2" = Table.AddColumn(#"Added Custom", "EndOfMonth", each Date.EndOfMonth([Date])),
12    #"Added Custom26" = Table.AddColumn(#"Added Custom25", "Month Year", each Date.ToText([Date], "MM-yyyy")),
13    #"Added Custom27" = Table.AddColumn(#"Added Custom26", "Month Year Sort", each Date.Year([Date])*100 + Date.Month([Date])),
14    #"Added Custom1" = Table.AddColumn(#"Added Custom27", "StartOfWeek", each Date.StartOfWeek([Date],1)),
15    #"Added Custom35" = Table.AddColumn(#"Added Custom1", "EndOfWeek", each Date.EndOfWeek([Date],1)),
16    #"Added Custom2" = Table.AddColumn(#"Added Custom35", "StartOfYear", each Date.StartOfYear([Date])),
17    #"Added Custom3" = Table.AddColumn(#"Added Custom2", "EndOfYear", each Date.EndOfYear([Date])),
18    #"Added Custom31" = Table.AddColumn(#"Added Custom3", "End Of Qtr", each Date.EndOfQuarter([Date])),
19    #"Added Custom32" = Table.AddColumn(#"Added Custom31", "Start Of Qtr", each Date.StartOfQuarter([Date])),
20    #"Added Custom33" = Table.AddColumn(#"Added Custom32", "Qtr Year", each Text.Combine({"Q", Number.ToText(Date.QuarterOfYear([Date])), "-", Number.ToText(Date.Year([Date]))})),
21    #"Added Custom34" = Table.AddColumn(#"Added Custom33", "Year Otr", each Date.Year([Date])*10 + Date.QuarterOfYear([Date])),
22    #"Added Custom4" = Table.AddColumn(#"Added Custom34", "Day of Week (Tue)", each Date.DayOfWeek([Date],2)>1),
23    #"Added Custom5" = Table.AddColumn(#"Added Custom4", "DayOfWeekName", each Date.DayOfWeekName([Date])),
24    #"Added Custom6" = Table.AddColumn(#"Added Custom5", "DayOfYear", each Date.DayOfYear([Date])),
25    #"Added Custom7" = Table.AddColumn(#"Added Custom6", "Add days", each Date.AddDays([Date],-3)),
26    #"Added Custom8" = Table.AddColumn(#"Added Custom7", "Add Months", each Date.AddMonths([Date],2)),
27    #"Added Custom9" = Table.AddColumn(#"Added Custom8", "Add Days", each Date.AddDays([Date],1)),
28    #"Added Custom10" = Table.AddColumn(#"Added Custom9", "Tried local now", each DateTimeZone.FixedLocalNow()),
29    #"Added Custom11" = Table.AddColumn(#"Added Custom10", "Today's date", each DateTime.Date( DateTime.FixedLocalNow())),
30    #"Added Custom12" = Table.AddColumn(#"Added Custom11", "Is In Current Day", each if Date.InCurrentDay([Date]) then "Today" else Date.ToText([Date])),
31    #"Added Custom13" = Table.AddColumn(#"Added Custom12", "Is Current Month", each Date.IsInCurrentMonth([Date])),
32    #"Added Custom14" = Table.AddColumn(#"Added Custom13", "Month Type", each if Date.IsInCurrentMonth([Date]) then "This Month"
33 else if Date.IsInNextMonth([Date]) then "Next Month"
34 else if Date.IsInPreviousMonth([Date]) then "Last Month" else Date.ToText([Date], "MM-yyyy")),
35    #"Added Custom16" = Table.AddColumn(#"Added Custom14", "FY Start", each if Date.Month([Date]) < 4 then #date(Date.Year([Date]), -1,4,1) else #date(Date.Year([Date]), 4,1)),

```

Back Cancel Next Publish

- Get Script from source: https://github.com/Huong2709/fabric/blob/main/create_date_table.txt

```

Huong2709 Add files via upload
Code Blame 54 lines (54 loc) + 7.34 KB
1 let
2     StartDate = #date(2012,01,01),
3     EndDate = Date.EndOfYear(Date.AddYears(Date.From(DateTime.FixedLocalNow()),1)),
4     Source = List.Dates(StartDate,
5         Duration.Days(EndDate-StartDate) +1,
6         #duration(1,0,0,0)),
7     #"Converted to Table" = Table.FromList(Source, Splitter.SplitByNothing(), null, null, ExtraValues.Error),
8     #"Renamed Columns" = Table.RenameColumns(#"Converted to Table",{"Column1", "Date"}),
9     #"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"Date", type date}}),
10    #"Added Custom" = Table.AddColumn(#"Changed Type", "StartOfMonth", each Date.StartOfMonth([Date])),
11    #"Added Custom9" = Table.AddColumn(#"Added Custom", "EndOfMonth", each Date.EndOfMonth([Date])),
12    #"Added Custom26" = Table.AddColumn(#"Added Custom25", "Month Year", each Date.ToText([Date], "MM-yyyy")),
13    #"Added Custom27" = Table.AddColumn(#"Added Custom26", "Month Year Sort", each Date.Year([Date])*100 + Date.Month([Date])),
14    #"Added Custom1" = Table.AddColumn(#"Added Custom27", "StartOfWeek", each Date.StartOfWeek([Date],1)),
15    #"Added Custom35" = Table.AddColumn(#"Added Custom1", "EndOfWeek", each Date.EndOfWeek([Date],1)),
16    #"Added Custom2" = Table.AddColumn(#"Added Custom35", "StartOfYear", each Date.StartOfYear([Date])),
17    #"Added Custom3" = Table.AddColumn(#"Added Custom2", "EndOfYear", each Date.EndOfYear([Date])),
18    #"Added Custom31" = Table.AddColumn(#"Added Custom3", "End Of Qtr", each Date.EndOfQuarter([Date])),
19    #"Added Custom32" = Table.AddColumn(#"Added Custom31", "Start Of Qtr", each Date.StartOfQuarter([Date])),
20    #"Added Custom33" = Table.AddColumn(#"Added Custom32", "Qtr Year", each Text.Combine({"Q", Number.ToText(Date.QuarterOfYear([Date])), "-", Number.ToText(Date.Year([Date]))})),
21    #"Added Custom34" = Table.AddColumn(#"Added Custom33", "Year Otr", each Date.Year([Date])*10 + Date.QuarterOfYear([Date])),
22    #"Added Custom4" = Table.AddColumn(#"Added Custom34", "Day of Week (Tue)", each Date.DayOfWeek([Date],2)>1),
23    #"Added Custom5" = Table.AddColumn(#"Added Custom4", "DayOfWeekName", each Date.DayOfWeekName([Date])),
24    #"Added Custom6" = Table.AddColumn(#"Added Custom5", "DayOfYear", each Date.DayOfYear([Date])),
25    #"Added Custom7" = Table.AddColumn(#"Added Custom6", "Add days", each Date.AddDays([Date],-3)),
26    #"Added Custom8" = Table.AddColumn(#"Added Custom7", "Add Months", each Date.AddMonths([Date],2)),
27    #"Added Custom9" = Table.AddColumn(#"Added Custom8", "Add Days", each Date.AddDays([Date],1)),
28    #"Added Custom10" = Table.AddColumn(#"Added Custom9", "Tried local now", each DateTimeZone.FixedLocalNow()),
29    #"Added Custom11" = Table.AddColumn(#"Added Custom10", "Today's date", each DateTime.Date( DateTime.FixedLocalNow())),
30    #"Added Custom12" = Table.AddColumn(#"Added Custom11", "Is In Current Day", each if Date.InCurrentDay([Date]) then "Today" else Date.ToText([Date])),
31    #"Added Custom13" = Table.AddColumn(#"Added Custom12", "Is Current Month", each Date.IsInCurrentMonth([Date])),
32    #"Added Custom14" = Table.AddColumn(#"Added Custom13", "Month Type", each if Date.IsInCurrentMonth([Date]) then "This Month"
33 else if Date.IsInNextMonth([Date]) then "Next Month"
34 else if Date.IsInPreviousMonth([Date]) then "Last Month" else Date.ToText([Date], "MM-yyyy")),
35    #"Added Custom16" = Table.AddColumn(#"Added Custom14", "FY Start", each if Date.Month([Date]) < 4 then #date(Date.Year([Date]), -1,4,1) else #date(Date.Year([Date]), 4,1)),

```

- Input script in M language

- Rename the table after creation if need

- Get data > More : to add data from other sources

Loading files from other sources

The screenshot shows the Power BI Dataflow interface with the 'Dataflow 1' tab selected. In the top left, there's a sidebar with various icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and a Dataflow 1 section. The main area is titled 'Power Query' and shows the 'Get data' step. It has a 'Choose data source' section with a search bar and a grid of options like Excel workbook, SQL Server database, SharePoint folder, Text/CSV, Power BI dataflows (Legacy), Dataflows, etc. Below that is an 'Upload file' section with a 'Browse...' button. At the bottom right of the main area are 'Cancel', 'Step', and 'Publish' buttons.

This screenshot shows the 'Transform' step of the Power BI Dataflow. The interface is similar to the 'Get data' step, with a sidebar on the left and a main 'Power Query' area. The main area shows a large table preview with many columns, including Date, StartOfMonth, EndOfMonth, MonthYear, MonthYearSort, StartOfWeek, EndOfWeek, StartOfYear, EndOfYear, StartQtr, EndQtr, and QtYear. The table contains 30 rows of data from January 2012 to March 2012. On the right side, there are 'Query settings' and 'Applied steps' sections, both currently empty. At the bottom right are 'Step', 'Publish', and a 'More...' button.

- Let's test with excel files
- Previously I can upload excel files from github, but recently I recently I have some issue with connection file by links. If you have same issue, please try to download the file to your computer and upload manually
- In this example, you can get the file from : https://github.com/Huong2709/fabric/blob/main/Sales_Data.xlsx

Get data

Connect to data source

Excel workbook
File
Learn more

Connection settings

Link to file Upload file (Preview)

Sales_Data 3.xlsx Upload successful (1.7 MB)

Connection credentials

Connection: [dropdown] Success Edit connection

Authentication kind: Organizational account Edit connection

Back Cancel Next

- We can upload files by using link or Upload from local file locations
- Select Connection settings
- Upload file or adding link to load files
- Click next

Dataflow 1 | No label

Power Query

Home Success

Get data

Choose data

Search

Display options

Excel workbook [4]

Customer Geography Item Sales

| OrderNo | ItemID | SalesDate | DeliveryDate | CustomerId | CityId | Qty | Price | Cost | DiscountPercent |
|---------|--------|-----------|--------------|------------|--------|-----|-------|------|-----------------|
| 20277 | 37 | 10/3/2020 | 10/6/2020 | 2691 | 26 | 1 | 233 | 172 | 12 |
| 22328 | 23 | 10/3/2020 | 10/7/2020 | 6085 | 61 | 1 | 60 | 54 | 9 |
| 22784 | 33 | 10/3/2020 | 10/9/2020 | 3562 | 19 | 1 | 333 | 259 | 5 |
| 25881 | 14 | 10/3/2020 | 10/9/2020 | 338 | 52 | 2 | 753 | 301 | 39 |
| 26399 | 52 | 10/3/2020 | 10/13/2020 | 4995 | 51 | 2 | 53 | 45 | 28 |
| 28885 | 51 | 10/3/2020 | 10/10/2020 | 2651 | 33 | 2 | 73 | 36 | 36 |
| 29167 | 33 | 10/3/2020 | 10/13/2020 | 1002 | 51 | 1 | 267 | 202 | 45 |
| 20273 | 32 | 10/2/2020 | 10/9/2020 | 4786 | 73 | 2 | 91 | 77 | 20 |
| 20724 | 46 | 10/2/2020 | 10/7/2020 | 5157 | 4 | 1 | 93 | 65 | 27 |
| 21121 | 53 | 10/2/2020 | 10/5/2020 | 386 | 33 | 2 | 233 | 172 | 10 |
| 21158 | 14 | 10/2/2020 | 10/11/2020 | 1162 | 69 | 1 | 65 | 52 | 32 |
| 21320 | 32 | 10/2/2020 | 10/10/2020 | 1879 | 62 | 2 | 40 | 36 | 13 |
| 22561 | 26 | 10/2/2020 | 10/3/2020 | 2954 | 32 | 2 | 83 | 66 | 29 |
| 23121 | 53 | 10/2/2020 | 10/10/2020 | 2025 | 93 | 2 | 91 | 77 | 25 |
| 23180 | 20 | 10/2/2020 | 10/10/2020 | 882 | 27 | 1 | 47 | 39 | 1 |
| 25941 | 40 | 10/2/2020 | 10/4/2020 | 3869 | 77 | 1 | 40 | 38 | 13 |
| 26619 | 29 | 10/2/2020 | 10/7/2020 | 1070 | 69 | 2 | 91 | 77 | 2 |
| 29171 | 49 | 10/2/2020 | 10/3/2020 | 1763 | 97 | 2 | 97 | 67 | 44 |
| 29189 | 17 | 10/2/2020 | 10/5/2020 | 5083 | 63 | 2 | 47 | 39 | 45 |
| 29191 | 39 | 10/2/2020 | 10/8/2020 | 4356 | 72 | 2 | 67 | 53 | 14 |
| 29236 | 32 | 10/2/2020 | 10/12/2020 | 1125 | 30 | 1 | 83 | 70 | 45 |
| 29969 | 15 | 10/2/2020 | 10/11/2020 | 441 | 63 | 2 | 267 | 202 | 12 |
| 20994 | 31 | 10/1/2020 | 10/9/2020 | 2004 | 83 | 1 | 333 | 259 | 12 |
| 21021 | 33 | 10/1/2020 | 10/6/2020 | 3651 | 45 | 2 | 83 | 70 | 18 |
| 21025 | 18 | 10/1/2020 | 10/8/2020 | 4123 | 96 | 1 | 40 | 38 | 5 |
| 21449 | 19 | 10/1/2020 | 10/5/2020 | 5592 | 35 | 2 | 65 | 52 | 36 |
| 21926 | 55 | 10/1/2020 | 10/1/2020 | 6017 | 86 | 2 | 200 | 140 | 9 |

Completed (8.59 s) Columns: 38 Rows: 99+

Back Cancel Create Step Publish

- Select related sheet in excel files what you want to upload as data source
- Click **Create**

Power Query Home Transform Add column View Help

Column from examples ▾ Custom function ▾ Invoke custom function Conditional column Rank column Duplicate column General From text From number Date Time Duration Date and time column

Queries [5] Custom column

| OrderNo | ItemID | SalesDate | DeliveryDate | CustomerID | CityID | Qty | Price | Cost | DiscountPercent |
|---------|--------|-----------|--------------|------------|--------|-----|-------|------|-----------------|
| 20277 | 37 | 10/3/2020 | 10/5/2020 | 2691 | 26 | 1 | 233 | 172 | 12 |
| 23238 | 23 | 10/3/2020 | 10/7/2020 | 6085 | 61 | 1 | 60 | 54 | 9 |
| 22784 | 53 | 10/3/2020 | 10/9/2020 | 3562 | 19 | 1 | 333 | 259 | 5 |
| 25881 | 14 | 10/3/2020 | 10/9/2020 | 338 | 52 | 2 | 753 | 301 | 39 |
| 26399 | 52 | 10/3/2020 | 10/13/2020 | 4995 | 51 | 2 | 53 | 45 | 28 |
| 28895 | 51 | 10/3/2020 | 10/10/2020 | 7661 | 22 | 2 | 78 | 56 | 26 |
| 29167 | 33 | 10/3/2020 | | | | | | | |
| 20273 | 32 | 10/2/2020 | | | | | | | |
| 20726 | 46 | 10/2/2020 | | | | | | | |
| 21121 | 53 | 10/2/2020 | | | | | | | |
| 21156 | 14 | 10/2/2020 | | | | | | | |
| 21320 | 32 | 10/2/2020 | | | | | | | |
| 22561 | 26 | 10/2/2020 | | | | | | | |
| 23121 | 53 | 10/2/2020 | | | | | | | |
| 23186 | 20 | 10/2/2020 | | | | | | | |
| 25941 | 40 | 10/2/2020 | | | | | | | |
| 26619 | 29 | 10/2/2020 | | | | | | | |
| 29171 | 49 | 10/2/2020 | | | | | | | |
| 29189 | 17 | 10/2/2020 | | | | | | | |
| 29191 | 39 | 10/2/2020 | | | | | | | |
| 29236 | 32 | 10/2/2020 | | | | | | | |
| 29969 | 15 | 10/2/2020 | | | | | | | |
| 20994 | 31 | 10/1/2020 | | | | | | | |
| 21021 | 33 | 10/1/2020 | | | | | | | |
| 21025 | 18 | 10/1/2020 | | | | | | | |
| 21449 | 19 | 10/1/2020 | | | | | | | |
| 21926 | 55 | 10/1/2020 | | | | | | | |
| 21928 | 19 | 10/1/2020 | | | | | | | |
| 22111 | 35 | 10/1/2020 | 10/10/2020 | 1909 | 80 | 2 | 93 | 65 | 29 |
| 22258 | 17 | 10/1/2020 | 10/6/2020 | 2220 | 39 | 2 | 97 | 67 | 18 |

Columns: 10 Rows: 99+

Similar as in Power BI, we can do similar activities in MS Fabric

Let's try to add 1 column as an example

- Click Customer column in action pane
- Input "New column name":
- Input Custom column formula
- Click OK

Custom column dialog:

New column name * **GROSS** Data type

Custom column formula * **= [Qty]*[Price]**

Available column(s) **DeliveryDate** **CustomerID** **CityID** **Qty** **Price** **Cost** **DiscountPercent**

Insert column

Learn more about Power Query formulas

OK Cancel

Power Query Home Transform Add column View Help

Column from examples ▾ Custom function ▾ Invoke custom function Conditional column Rank column Duplicate column General From text From number Date Time Duration Date and time column

Queries [5] Custom column

| Date | StartOfMonth | EndOfMonth | MonthYear | MonthYearSort | StartOfWeek | EndOfWeek | StartOfWeek | EndOfWeek | StartOfYear | EndOfYear | StartOfQtr | EndOfQtr | StartOfYr | EndOfYr |
|-----------|-----------------------|------------------------|-----------|---------------|-------------------------|-----------------------|-----------------------|-------------------------|-----------------------|------------------------|------------|----------|-----------|---------|
| 1/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 12/26/2011, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 2/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 3/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 4/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 5/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 6/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 7/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 8/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 9/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 10/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 11/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 12/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 13/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 14/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 15/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 16/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 17/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 18/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 19/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 20/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 21/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 22/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 23/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 24/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 25/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 26/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 27/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 28/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |
| 29/1/2012 | 1/1/2012, 12:00:00 AM | 1/31/2012, 12:00:00 AM | Jan-2012 | 201201 | 1/2/2012, 12:00:00 AM | 1/8/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 12/31/2012, 12:00:00 AM | 1/1/2012, 12:00:00 AM | 3/31/2012, 12:00:00 AM | Q1-2012 | | | |

Query settings

- Name: **Date**
- Entity type: **Custom**
- Applied steps
- Added Column 1
- Added Column 2
- Added Column 3
- Added Column 4
- Added Column 5
- Added Column 6
- Added Column 7
- Added Column 8
- Added Column 9
- Added Column 10
- ABC Changed...
- Rename...

Data destination

- Lakehouse 1

Step Step

Connect to data destination

The screenshot shows the 'Connection credentials' step of a data connection setup. At the top left is a 'Lakehouse' icon with the text 'Lakehouse BETA Microsoft Fabric (Preview)'. To the right is a 'Connection' dropdown menu set to 'Lakehouse (none)' with a refresh icon. Below it is a note 'Authentication kind: Organizational account' and a link 'Edit connection'. At the bottom right are 'Cancel' and 'Next' buttons.

Select Lakehouse connection

Choose destination target

The screenshot shows the 'Choose destination target' step. It features a search bar, 'Display options' dropdown, and a tree view of workspace structures. The tree includes 'Fabric1', 'Fabric2', 'Fabric3', 'Fabric4' (with a [2] indicator), 'DataflowsStagingLakehouse', 'lakehouse1' (which is selected and highlighted in grey), and 'My workspace'. A note says 'A new table will be created in lakehouse1'. A 'Table name' input field contains 'Date'. At the bottom are 'Back' and 'Next' buttons.

- In Display option, navigate to Workspace > lakehouse that we want to load data into
- Click next

Choose destination settings

Update method



Column mapping

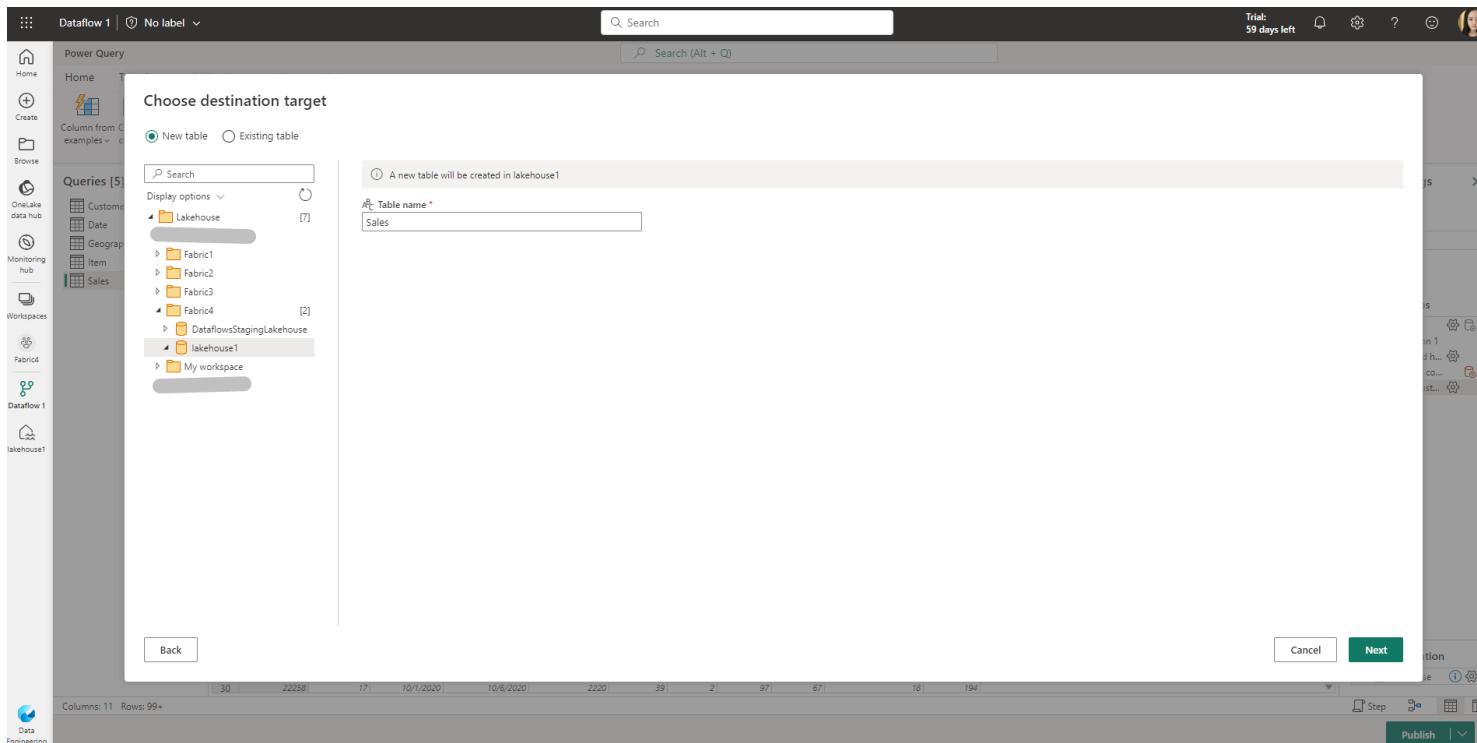
| Source | Source type | Destination | Destination type |
|-----------------|------------------|-----------------|------------------|
| FYDayofYear | 1-3 Whole number | FYDayofYear | Whole number |
| FyMonth | 1-3 Whole number | FyMonth | Whole number |
| WeekofYear | 1-3 Whole number | WeekofYear | Whole number |
| YearWeek | 1-3 Whole number | YearWeek | Whole number |
| FYWeekYearStart | Date/Time | FYWeekYearStart | Date/Time |
| FYWeek | 1-3 Whole number | FYWeek | Whole number |
| FYYearWeek | 1-3 Whole number | FYYearWeek | Whole number |
| FYQtrNo | 1-3 Whole number | FYQtrNo | Whole number |
| FYQtrStartDate | Date/Time | FYQtrStartDate | Date/Time |
| FYEndofQtr | Date/Time | FYEndofQtr | Date/Time |
| FYQtr | 1-3 Whole number | FYQtr | Whole number |
| QtrFY | Text | QtrFY | Text |

Back

Cancel

Save settings

- Drill up and down to check Column mapping details, source type then Save setting



The screenshot shows the 'Choose destination target' dialog box in Power Query. The 'New table' radio button is selected. A new table will be created in the 'lakehouse1' fabric. The table name is set to 'Sales'. The background shows the Power Query interface with a list of queries and a workspace sidebar.

Continuing with sales table

- Navigate to Workspace > lakehouse that we want to load data into

| Source | Source type | Destination | Destination type |
|-----------------|--------------|-----------------|------------------|
| OrderNo | Whole number | OrderNo | Whole number |
| ItemID | Whole number | ItemID | Whole number |
| SalesDate | Date | SalesDate | Date |
| DeliveryDate | Date | DeliveryDate | Date |
| CustomerID | Whole number | CustomerID | Whole number |
| CityID | Whole number | CityID | Whole number |
| Qty | Whole number | Qty | Whole number |
| Price | Whole number | Price | Whole number |
| Cost | Whole number | Cost | Whole number |
| DiscountPercent | Whole number | DiscountPercent | Whole number |
| Gross | Whole number | Gross | Whole number |

- For custom column that we created in power query, It doesn't pop up data type and select column mapping. Make sure to double check before "Save setting" else you can't see it in Lakehouse after publish

Completed (3.71 s) Columns: 11 Rows: 99+

Query settings

Name: Sales

Entity type: Custom

Applied steps

- Source
- Navigation 1
- Promoted h...
- ABC Changed co...
- Added cust...
- ABC Changed co...

Data destination

- Lakehouse

>> After publish for sales table, please continue to setting Lakehouse and publish for other tables (Customer, Geography, Item)

1.4 Using SQL endpoint

After adding at least 1 table in lakehouse, SQL is automatically enable for using

SQL endpoint operates in read-only mode over lakehouse delta tables

A screenshot of the Azure Data Lake Storage Gen2 Home screen. On the left, there's a sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and lakehouse1. The main area shows an 'Explorer' view with a tree structure under 'lakehouse1' showing 'Tables' (Customer, Date, Geography, Item, Sales) and 'Files'. A message at the top says: 'A SQL endpoint for SQL querying and a default dataset for reporting were created and will be updated with any tables added to the lakehouse.' To the right, there's a 'Get data in your lakehouse' section with four buttons: 'New Dataflow Gen2', 'New data pipeline', 'Open notebook', and 'New shortcut'. In the top right corner, there's a 'Lakehouse' button with the sub-options 'Explore your data files and folders' and 'SQL endpoint Query data using SQL'. A yellow box highlights the 'SQL endpoint' button.

- From Lakehouse screen > Click Lakehouse icon> Click SQL endpoint

A screenshot of the Azure Data Lake Storage Gen2 Home screen, similar to the previous one but with a different message at the top: 'A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the lakehouse. Learn more' with a 'Manage default Power BI dataset' button. The 'Lakehouse' icon in the top right is highlighted with a yellow box. The rest of the interface is identical to the first screenshot, showing the Explorer view, Get data options, and the 'SQL endpoint' button.

The screenshot shows the Azure Data Lake Studio interface. On the left, there's a sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and lakehouse1 (which is selected). The main area has a search bar at the top. Below it, there are tabs for Home, New SQL query, and New visual query. A message indicates a default dataset was created for faster reporting. The Explorer sidebar shows a tree structure with Warehouses, lakehouse1 (expanded), and Queries (also expanded) containing My queries, SQL query 1, and SQL query 2. The SQL query 2 tab is active, showing a query script:

```
1 --SQL Query 2
2 SELECT a2.Category, YEAR(a3.Date) as Year, SUM(Gross) as Gross
3 FROM Sales a1
4 LEFT JOIN Item a2 ON a1.ItemID=a2.ItemID
5 LEFT JOIN Date a3 ON a1.SalesDate=a3.Date
6 GROUP BY a2.Category, YEAR(a3.Date)
7 ORDER BY a2.Category, YEAR(a3.Date)
```

The Results tab is selected, displaying a table with the following data:

| | ABC Category ↑ | 123 Year | 12L Gross |
|----|----------------|----------|-----------|
| 1 | Category 1 | 2018 | 686549 |
| 2 | Category 1 | 2019 | 1355753 |
| 3 | Category 1 | 2020 | 260570 |
| 4 | Category 2 | 2018 | 1059609 |
| 5 | Category 2 | 2019 | 1903994 |
| 6 | Category 2 | 2020 | 371487 |
| 7 | Category 3 | 2018 | 448119 |
| 8 | Category 3 | 2019 | 828956 |
| 9 | Category 3 | 2020 | 171596 |
| 10 | Category 4 | 2018 | 765536 |
| 11 | Category 4 | 2019 | 1390501 |
| 12 | Category 4 | 2020 | 233183 |
| 13 | Category 5 | 2018 | 234802 |
| 14 | Category 5 | 2019 | 481669 |
| 15 | Category 5 | 2020 | 86484 |

At the bottom, a status message says "Succeeded (2 sec 586 ms)".

1.5 Create model

Model is for creation connections between tables so we can create reports that getting data from many tables

lakehouse1 | No label

Search

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

lakehouse1

Data Engineering

lakehouse1

Home

New SQL query New visual query

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the lakehouse. [Learn more](#)

Explorer

+ Warehouses

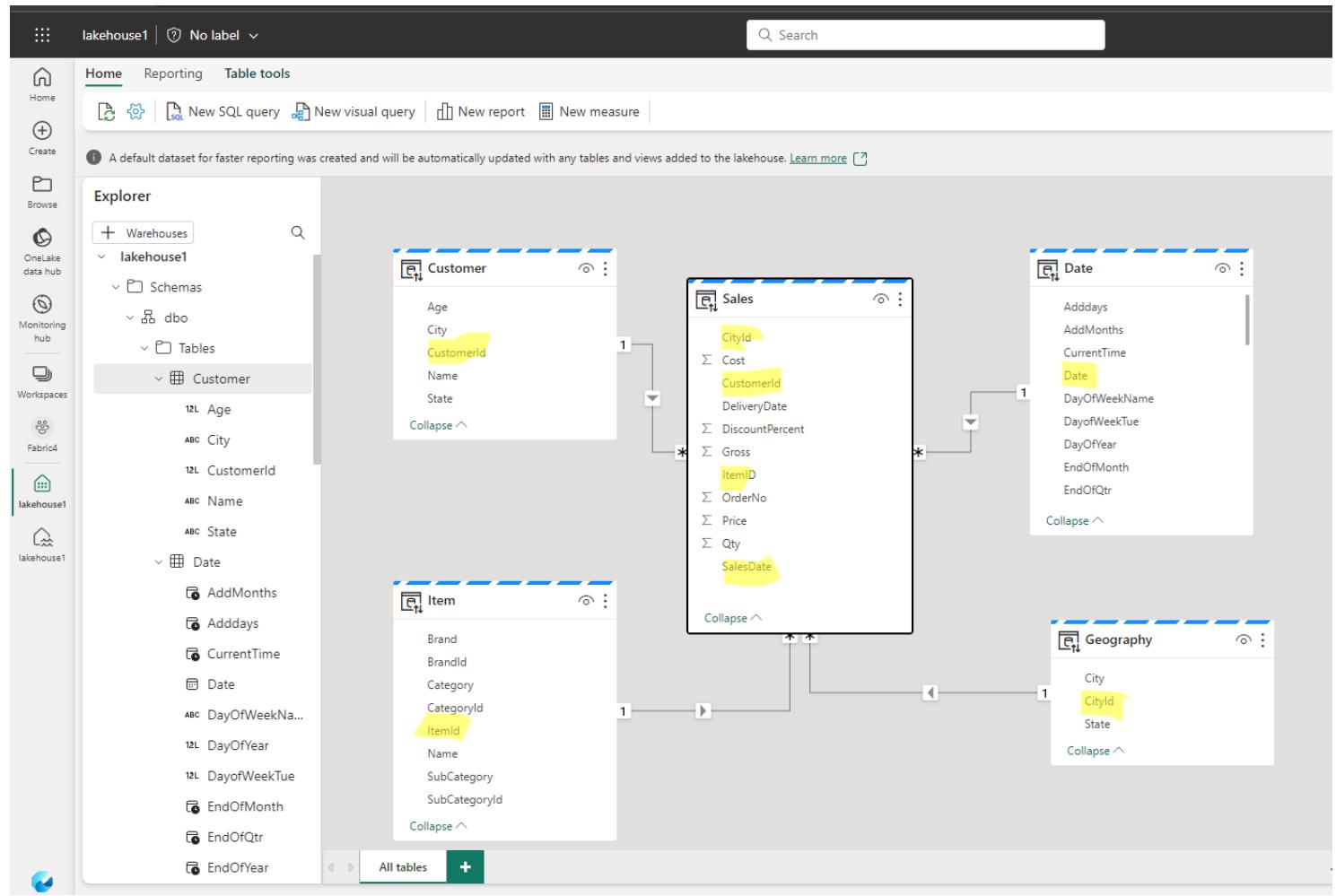
lakehouse1

- Schemas
 - dbo
 - Tables
 - Customer
 - Date
 - Geography
 - Item
 - Sales
 - Views
 - Functions
 - Stored Procedur...
 - guest
 - INFORMATION_SCHE...
 - sys
 - Security
- Queries
 - My queries
 - SQL query 1
 - SQL query 2
 - Shared queries

Data Query Model

The screenshot shows the Databricks interface with the 'lakehouse1' workspace selected. The left sidebar contains navigation links for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and the current workspace 'lakehouse1'. The main area displays the 'Explorer' sidebar, which lists the schema and query sections of the 'lakehouse1' warehouse. The 'Model' tab at the bottom of the sidebar is highlighted with a yellow box. A search bar is located at the top right.

Drag and drop between tables to create relationships base on key

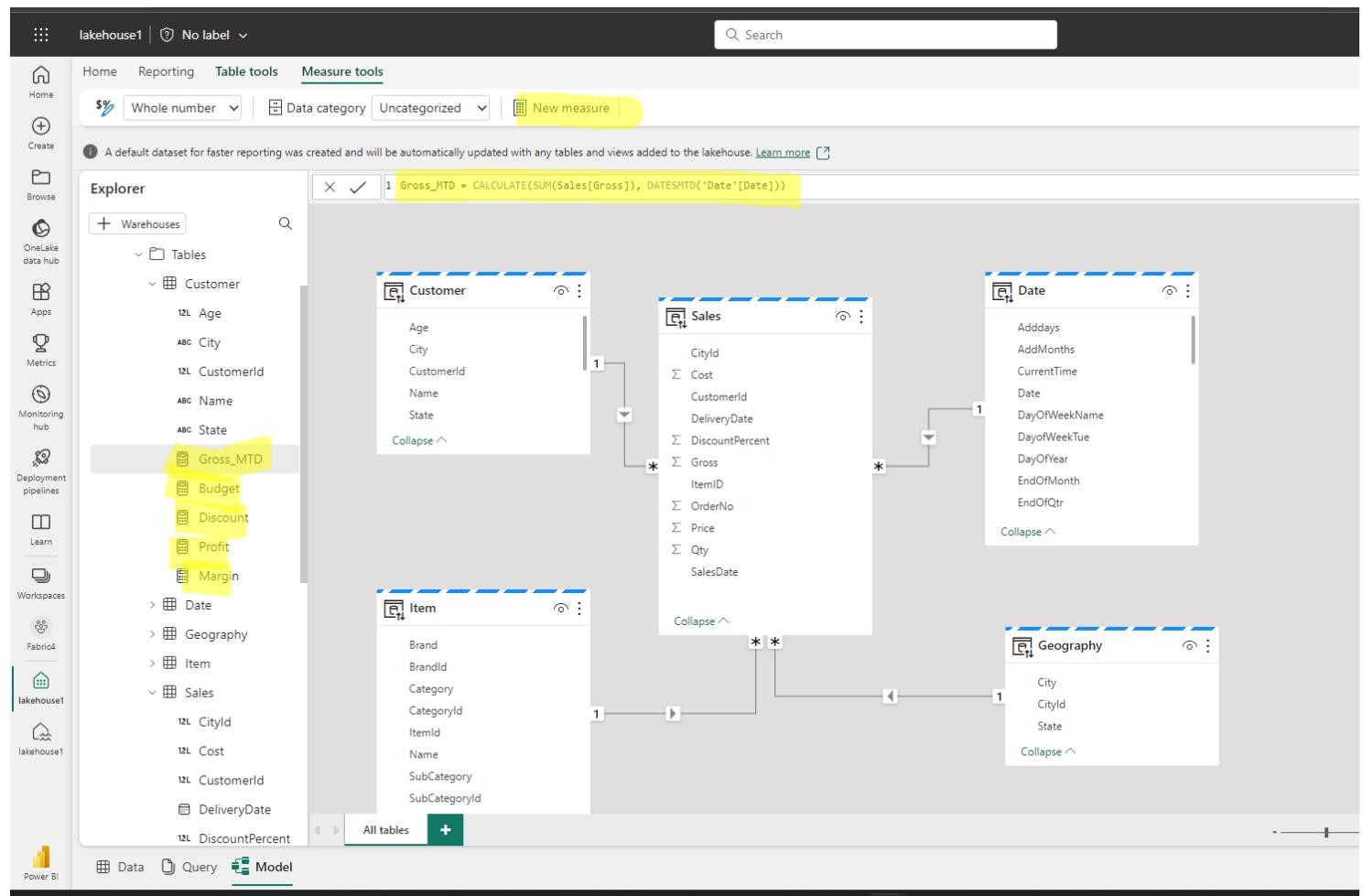


Sales table is including transaction. It's fact table, data is updated frequently. Let's put it in the central of model for easy viewing while creation relationships

Customer, item, date, Geography are dimension tables (Dim)

1.6 Create measure for reports

Measures are used in some of the most common data analyses. Simple summarizations such as sums, averages, minimum, maximum and counts can be set through the Fields well.



Sample script for testing measure:

```

Gross_MTD = CALCULATE(SUM(Sales[Gross]), DATESMTD('Date'[Date]))
Budget = SUMX(Sales, [Price] * [Qty])
Discount = SUMX(Sales, [Gross] * [DiscountPercent]/100)
Profit = SUMX(Sales, [Gross] - [Discount] - [Budget])
Margin = DIVIDE([Profit], SUMX(Sales, [Gross]))

```

- Click New measure
- Input formular for measure (wait 5-10 seconds for Fabric saving formular, else it will reloading the page)

1.7 Create Power BI reports

We can open report design platform like power BI in MS fabric

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the lakehouse. [Learn more](#)

Explorer

- + Warehouses
- Tables
 - Customer
 - Age
 - City
 - CustomerId
 - Name
 - State
 - Sales
 - CityId
 - Cost
 - CustomerId
 - DeliveryDate
 - DiscountPercent
 - Gross
 - ItemID
 - OrderNo
 - Price
 - Qty
 - SalesDate
 - Item
 - Brand
 - BrandId
 - Category
 - CategoryId
 - ItemID
 - Name
 - SubCategory
 - SubCategoryId
 - Date
 - AddDays
 - AddMonths
 - CurrentTime
 - Date
 - DayOfWeekName
 - DayOfWeekTue
 - DayOfYear
 - EndOfMonth
 - EndOfQuarter
 - Geography
 - City
 - CityId
 - State

All tables +

Data Query Model

- Click New report button

This is our result after drag and drop fields

Sales dashboard

Profit by State

Margin by EndOfMonth

Sum of Gross by Category

Sum of Gross by Brand

1. Select chart in Visualizations area

Visualizations

Filters

Search

Filters on this visual

- Category is (All)
- Profit is (All)

Add data fields here

Filters on this page

Add data fields here

Filters on all pages

Add data fields here

Details

Add data fields here

Values

Profit

Category

Category

Brand

BrandId

Category

Item

Date

Geography

Brand

BrandId

Category

ItemID

Name

SubCategory

SubCategoryID

Data

Customer

- Age
- Budget
- City
- CustomerId
- Discount
- Gross_MTD
- Margin
- Name
- State
- Profit
- State

Sales

Item

Date

Geography

2. Drag and drop relate data (Category , Profit)

>> Yeah, we can create a report. This can be done easily even we haven't learnt Power BI. You can try creating other reports for experience how cool MS Fabric is

Report

The screenshot shows the Microsoft Fabric interface with two main panes. On the left is the 'Explorer' pane, which lists datasets and tables. A red arrow points from the 'Customer' table in the 'Warehouses' section to the 'Sales dashboard' on the right. The 'Sales dashboard' contains several visualizations, including a map of North America, a line chart for 'Margin by EndOfMonth', and a stacked bar chart for 'Sum of Gross by Category'.

2. Notebook – Warehouse- pipeline

2.1 Create Notebook and LakeHouse from Workspace

The screenshot shows the Microsoft Fabric workspace with a 'New' workspace. A red arrow points to the 'Notebook (Preview)' icon. The 'Lakehouse explorer' pane shows a notebook named 'Notebook 1'. A modal dialog titled 'Add lakehouse' is open, with the 'New lakehouse' radio button selected. The 'Add' button is highlighted in green.

- Select workspace from MS Fabric home page

- Click Notebook icon

>> Wait some seconds till system open next from. I didn't know it so I keep clicking many times till more than 4 notebooks were created in my workspace

- After Welcome notebook screen is shown, click "Add" button to add Lakehouse. We can create new lakehouse or use existing lakehouse. In this example, let's create new lake house for testing

2.2 Adding data to LakeHouse from NoteBook

Now we can use notebook in MS Fabric

Check the below scrip source for Notebook experience

Link : https://github.com/Huong2709/fabric/blob/main/Notebook01_toLH04.ipynb

```
In [ 1]: import pandas as pd
In [ 1]: # Read data from GitHub
df = pd.read_csv('https://github.com/Huong2709/fabric/raw/main/Iris.csv')
df.head()
In [ 1]: df = df.iloc[:,5]
df.head()
In [ 1]: # write data to csv format file
df.to_csv('abfss://workspacell@onelake.dfs.fabric.microsoft.com/Lakehouse03.Lakehouse/Files/Github_Iris.csv', index=False)
In [ 1]: # write data to parquet format file
df.to_parquet('abfss://workspacell@onelake.dfs.fabric.microsoft.com/Lakehouse03.Lakehouse/Files/Github_Iris.parquet')

Create a managed table

In [ 1]: # Read data from parquet format file using spark
df2 = spark.read.parquet('abfss://workspacell@onelake.dfs.fabric.microsoft.com/Lakehouse03.Lakehouse/Files/Github_Iris.parquet')
df2.head()
In [ 1]: # write data to a table
df2.write.mode('overwrite').format('delta').saveAsTable('Iris_spark_internal')
```

You can write Python code yourself for copy code for check experience

NoteBook-Fabric4- LakeHouse10 | No label · Saved ▾

Search

Home Edit Run Data View

Spark jobs (2 of 2 succeeded) Log

Row(sepalLength=5.1, sepalWidth=3.5, petalLength=1.4, petalWidth=0.2, iris='Iris-setosa')

1 # Write data to a table
2 df2.write.mode('overwrite').format('delta').saveAsTable('Iris_spark_internal')
✓ 12 sec -Command executed in 12 sec 73 ms by Hien Tran on 9/07/33 PM, 8/19/23

Spark jobs (9 of 9 succeeded) Log

Create an external table

1 # Read data from csv format file using spark
2 df3 = spark.read.csv('abfss://Fabric4@onelake.dfs.fabric.microsoft.com/LakeHouse10.Lakehouse/Files/Github_Iris.csv', header=True)
3 df3.head()
✓ 1 sec -Command executed in 861 ms by Hien Tran on 9/07/34 PM, 8/19/23

> Spark Jobs (2 of 2 succeeded)

... Row(sepalLength='5.1', sepalWidth='3.5', petalLength='1.4', petalWidth='0.2', iris='Iris-setosa')

+ Code + Markdown

Note: Replace Your workspace and Lakehouse name in the script else it will throw error

The screenshot shows a Databricks notebook interface with the following details:

- Home Bar:** Home, Edit, Run, Data, View, Stop session, Language (PySpark (Python)), Open in VS Code.
- Lakehouse explorer:** Shows a tree view of LakeHouse[0] containing Tables (iris_cluster_kmean, iris_spark_external, iris_spark_external_2, iris_spark_external_3, iris_spark_internal) and Files (Iris_external).
- Code Editor:** Displays three code cells with their execution logs:
 - [5] # Write data to csv format file
df.to_csv('abfss://Fabric4@onelake.dfs.fabric.microsoft.com/LakeHouse10.Lakehouse/Files/Github_Iris.csv', index=False)
1 sec -Command executed in 923 ms by Hien Tran on 6:08:18 AM, 8/27/23
 - [6] # Write data to parquet format file
df.to_parquet('abfss://Fabric4@onelake.dfs.fabric.microsoft.com/LakeHouse10.Lakehouse/Files/Github_Iris.parquet')
2 sec -Command executed in 1 sec 578 ms by Hien Tran on 6:08:25 AM, 8/27/23
 - [7] # Read data from parquet format file using spark
df2 = spark.read.parquet('abfss://Fabric4@onelake.dfs.fabric.microsoft.com/LakeHouse10.Lakehouse/Files/Github_Iris.parquet')
df2.head()
2 sec -Command executed in 1 sec 667 ms by Hien Tran on 6:08:31 AM, 8/27/23
- Create a managed table:** A section titled "Create a managed table" with the following code:

```
1 # Write data to a table
2 df2.write.mode('overwrite').format('delta').saveAsTable('Iris_spark_internal')
```
- Status Bar:** Session ready, Save option: Automatic.

2.3 New NoteBook in Lakehouse

The screenshot shows the Databricks Home interface. At the top, there's a navigation bar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, Notebook 2, Lakehouse06, LakeHouse10 (selected), Warehouse01, and Data. The 'LakeHouse10' section is expanded, showing 'Tables' and 'Files'. Under 'Tables', 'iris_spark_external' is selected. The main area displays a table titled 'iris_spark_external' with columns: sepallength, sepalwidth, petallength, petalwidth, and iris. The table contains 150 rows of Iris flower data. A message at the top right says 'be updated with any tables added to the lakehouse.' Below the table, it says 'Showing 150'.

| | sepallength | sepalwidth | petallength | petalwidth | iris |
|----|-------------|------------|-------------|------------|-----------------|
| 1 | 4.8 | 3.0 | 1.4 | 0.1 | Iris-setosa |
| 2 | 4.3 | 3.0 | 1.1 | 0.1 | Iris-setosa |
| 3 | 4.8 | 3.0 | 1.4 | 0.3 | Iris-setosa |
| 4 | 5.9 | 3.0 | 4.2 | 1.5 | Iris-versicolor |
| 5 | 5.6 | 3.0 | 4.5 | 1.5 | Iris-versicolor |
| 6 | 6.6 | 3.0 | 4.4 | 1.4 | Iris-versicolor |
| 7 | 6.7 | 3.0 | 5.0 | 1.7 | Iris-versicolor |
| 8 | 5.4 | 3.0 | 4.5 | 1.5 | Iris-versicolor |
| 9 | 5.6 | 3.0 | 4.1 | 1.3 | Iris-versicolor |
| 10 | 6.1 | 3.0 | 4.6 | 1.4 | Iris-versicolor |
| 11 | 5.7 | 3.0 | 4.2 | 1.2 | Iris-versicolor |
| 12 | 7.1 | 3.0 | 5.9 | 2.1 | Iris-virginica |
| 13 | 6.5 | 3.0 | 5.8 | 2.2 | Iris-virginica |
| 14 | 7.6 | 3.0 | 6.6 | 2.1 | Iris-virginica |

Source code

https://github.com/Huong2709/fabric/blob/main/Notebook02_Iris_KMean.ipynb

```
In [ ]: import pandas as pd
from sklearn.cluster import KMeans

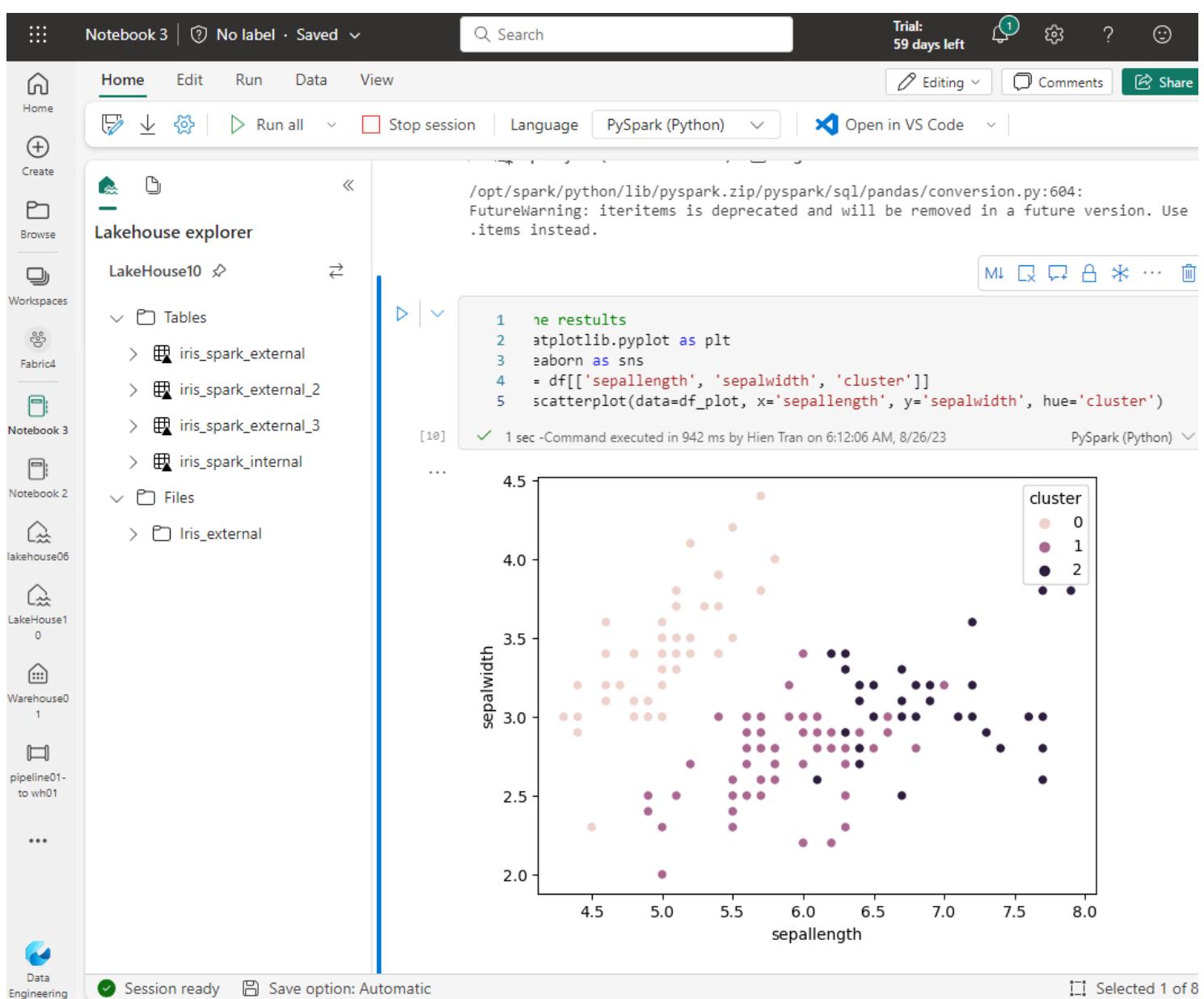
In [ ]: # Read data from file Github_Iris.csv in Workspace03/Lakehouse04
df = pd.read_csv('abfss://Workspace03@onelake.dfs.fabric.microsoft.com/Lakehouse04.Lakehouse/Files/Github_Iris.csv')
df.head()

In [ ]: # Choose features
column_for_cluster = ['sepallength', 'sepalwidth', 'petallength', 'petalwidth']
X = df[column_for_cluster]
X.head()

In [ ]: # Build clustering model
model_km = KMeans(n_clusters=3)
model_km.fit(X)

In [ ]: df['cluster'] = model_km.labels_
df.head()
```

Create Scatter chart by Notebook





Home



Create



Browse



Workspaces



Fabric4



Notebook 3



Notebook 2



Lakehouse06



LakeHouse10



Warehouse01

pipeline01-to wh01

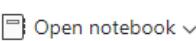


Data Engineering

Home



Get data ▾ New Power BI dataset



A SQL en

Create a new Power BI dataset with specific tables and views from the warehouse. Then save the dataset to your workspace for sharing, data modeling, and reporting.

ated and will be updated with any tables added to the lakehouse.

Explorer

LakeHouse10

Tables

iris_cluster_kmean

iris_spark_external

...

iris_spark_external_2

iris_spark_external_3

iris_spark_internal

Files

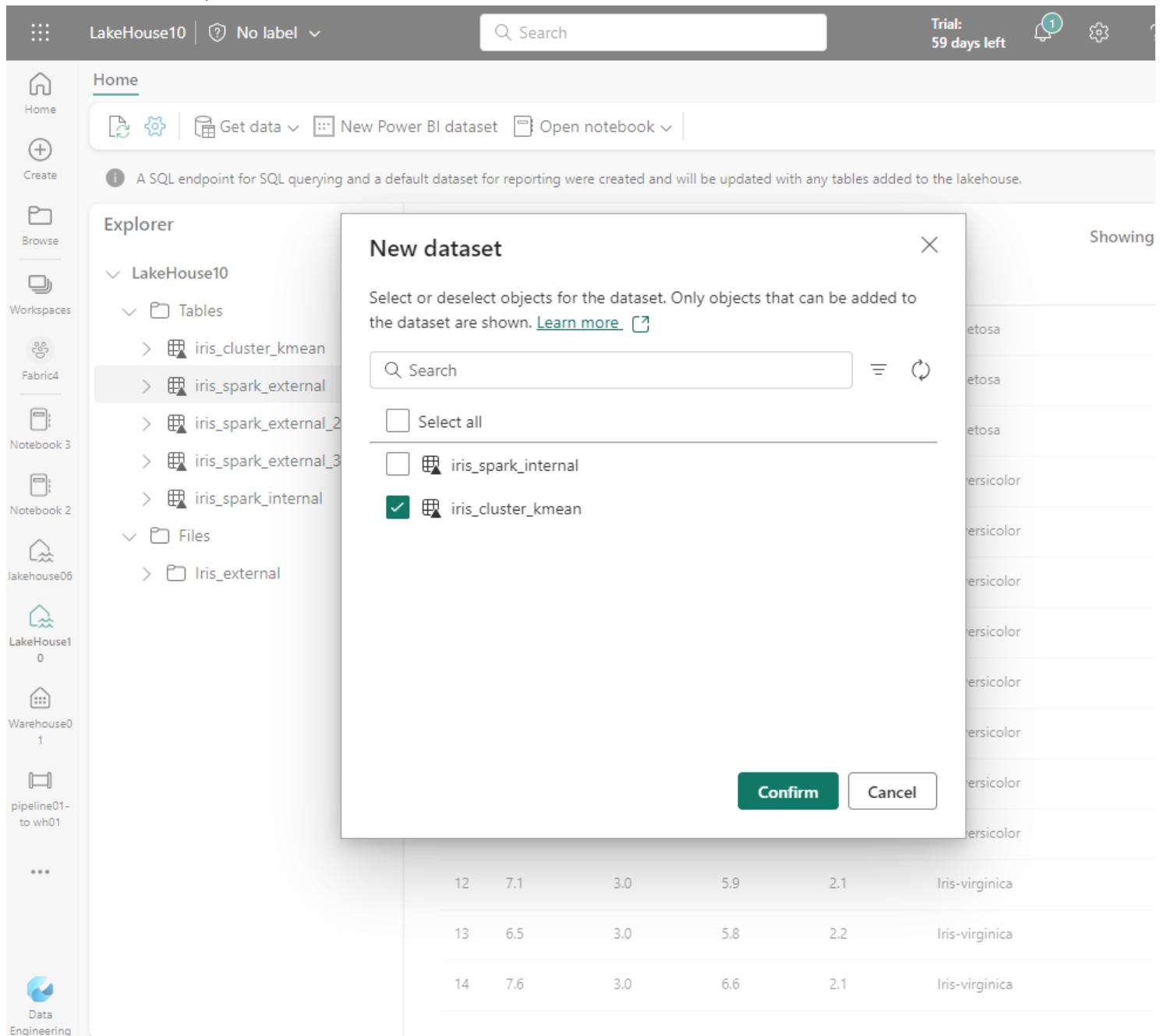
Iris_external

iris_spark_external

Showing 150 r

| | sepallength | sepalwidth | petallength | petalwidth | iris |
|----|-------------|------------|-------------|------------|-----------------|
| 1 | 4.8 | 3.0 | 1.4 | 0.1 | Iris-setosa |
| 2 | 4.3 | 3.0 | 1.1 | 0.1 | Iris-setosa |
| 3 | 4.8 | 3.0 | 1.4 | 0.3 | Iris-setosa |
| 4 | 5.9 | 3.0 | 4.2 | 1.5 | Iris-versicolor |
| 5 | 5.6 | 3.0 | 4.5 | 1.5 | Iris-versicolor |
| 6 | 6.6 | 3.0 | 4.4 | 1.4 | Iris-versicolor |
| 7 | 6.7 | 3.0 | 5.0 | 1.7 | Iris-versicolor |
| 8 | 5.4 | 3.0 | 4.5 | 1.5 | Iris-versicolor |
| 9 | 5.6 | 3.0 | 4.1 | 1.3 | Iris-versicolor |
| 10 | 6.1 | 3.0 | 4.6 | 1.4 | Iris-versicolor |
| 11 | 5.7 | 3.0 | 4.2 | 1.2 | Iris-versicolor |
| 12 | 7.1 | 3.0 | 5.9 | 2.1 | Iris-virginica |
| 13 | 6.5 | 3.0 | 5.8 | 2.2 | Iris-virginica |
| 14 | 7.6 | 3.0 | 6.6 | 2.1 | Iris-virginica |

Create Scatter chart by Power BI





Home



Create



Browse



Workspaces



Fabric4



Notebook 3



Notebook 2



lakehouse06

LakeHouse1
0Warehouse0
1pipeline01-
to wh01

Data
Engineering

Home

Help



New measure



New column



New table

Calculations



Manage roles



New report



Edit tables

Security

Reporting

Modeling

Properties

^ Cards

Show the database in the header when applicable

No

Show related fields when card is collapsed

Yes

Pin related fields to top of card

No

iris_cluster_kmean

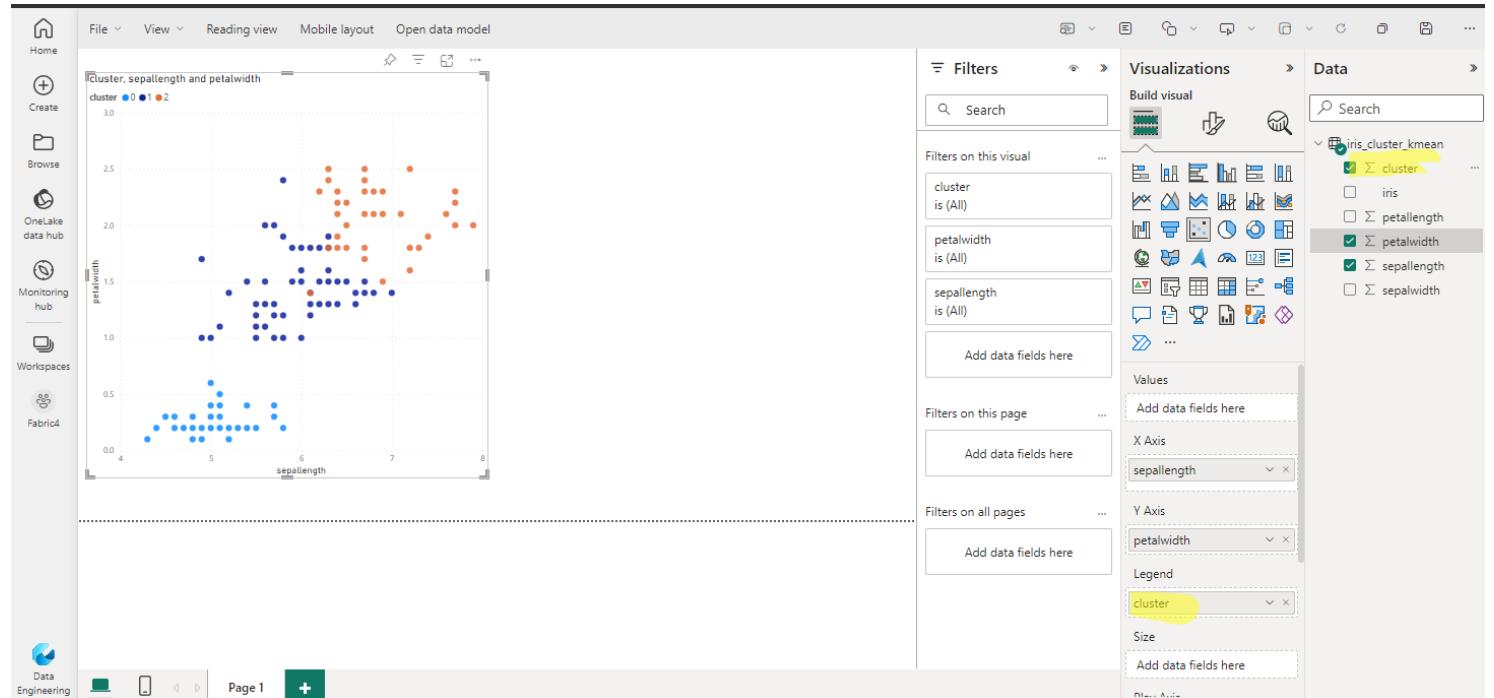
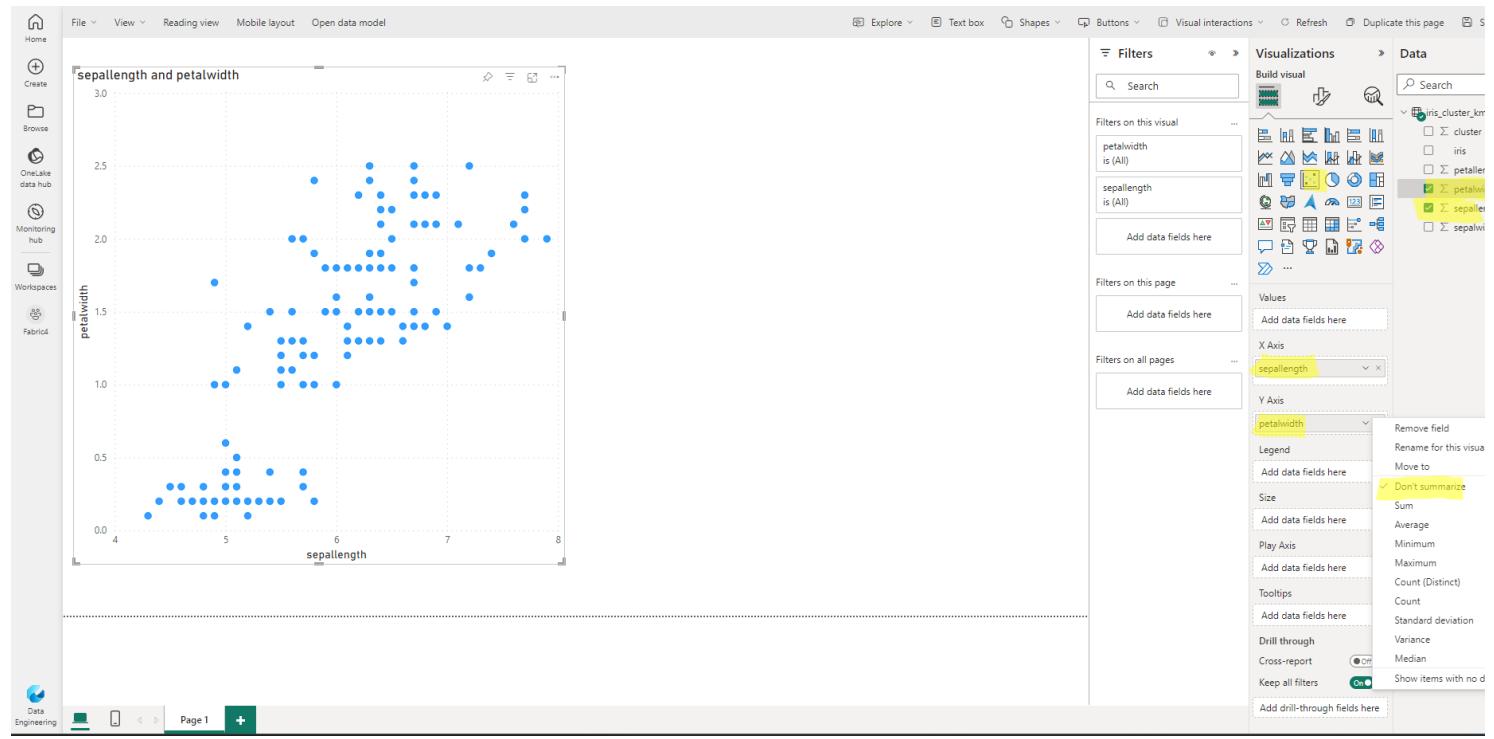
cluster
iris
petallength
petalwidth
sepallength
sepalwidth

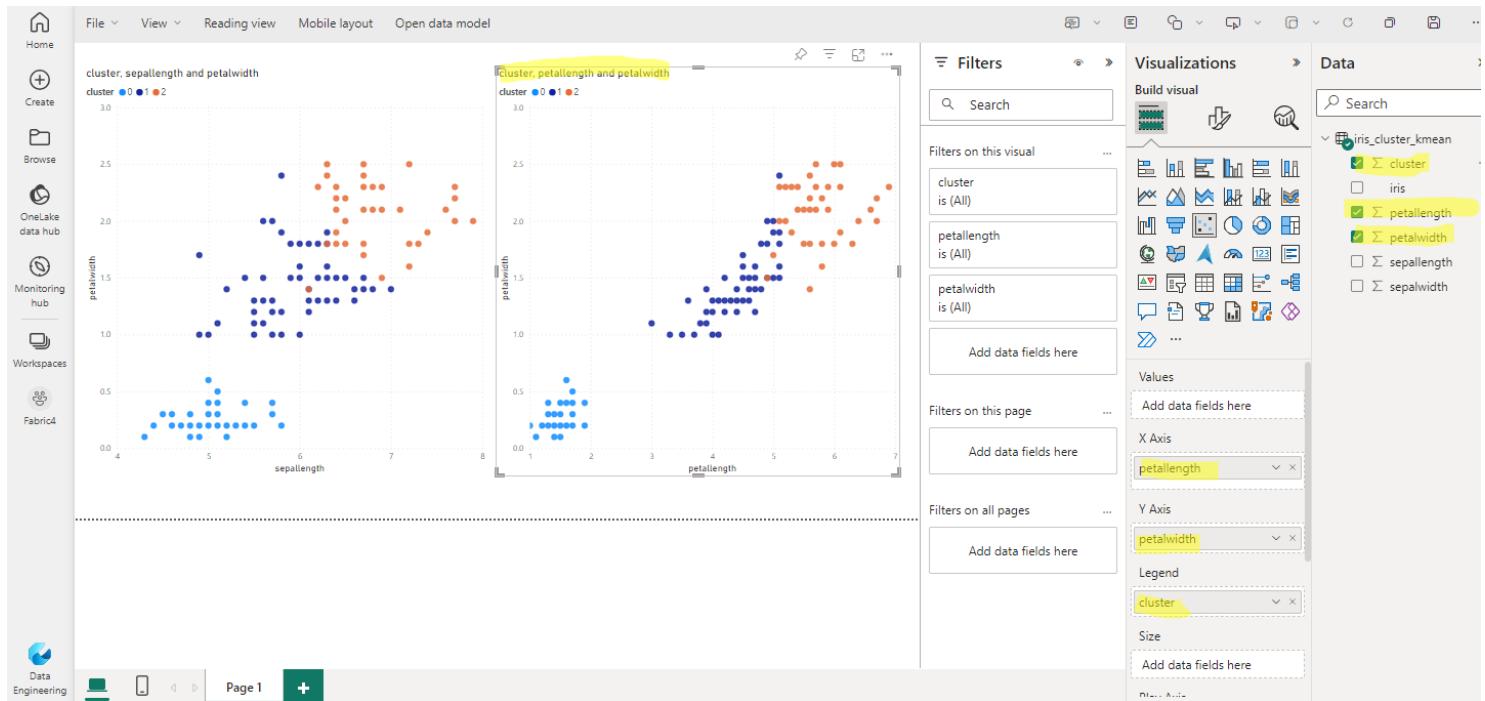
Collapse ^



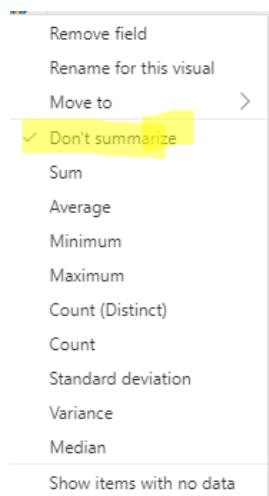
All tables







Note: Click in X Axis, Y Axis and select “Don’t summarize”



2.4 Create warehouse

The screenshot shows the Synapse Data Engineering workspace interface. On the left, there's a sidebar with various icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, NoteBook-Fabric4..., and Data Engineering. The main area has a title bar "Hitachi Solutions | Synapse Data Engineering Fabric4" and a search bar. A navigation bar at the top includes "New", "Upload", "Create deployment pipeline", "Update app", "Manage access", and "...". A modal window titled "Fabric4" is open, showing a list of items under "New": Data pipeline (Preview), Dataflow Gen2 (Preview), Eventstream (Preview), Experiment (Preview), KQL Database (Preview), KQL Queryset (Preview), Lakehouse (Preview), Model (Preview), Notebook (Preview), Report, Spark Job Definition (Preview), and Warehouse (Preview). The "Warehouse (Preview)" item is highlighted with a yellow background. A tooltip for "Warehouse (Preview)" states: "Provide strategic insights from multiple sources into your entire business." Below the modal, the main workspace shows a table of resources with columns: Type, Owner, Refreshed, and Next refresh. The table lists several entries: Dataflow Gen2 (Owner: Hien Tran, Refreshed: 8/15/23, 7:22:58 PM, Next refresh: N/A); Dataset (default) (Owner: Fabric4, Refreshed: 8/15/23, 7:07:47 PM, Next refresh: N/A); SQL endpoint (Owner: Fabric4, Refreshed: 8/15/23, 7:22:18 PM, Next refresh: N/A); Lakehouse (Owner: Hien Tran, Refreshed: —, Next refresh: —); Dataset (default) (Owner: Fabric4, Refreshed: 8/15/23, 7:07:49 PM, Next refresh: N/A); Warehouse (Owner: Hien Tran, Refreshed: —, Next refresh: N/A); Notebook (Owner: Hien Tran, Refreshed: —, Next refresh: —); Notebook (Owner: Hien Tran, Refreshed: —, Next refresh: —); Dataset (default) (Owner: Fabric4, Refreshed: 8/15/23, 7:07:14 PM, Next refresh: N/A); SQL endpoint (Owner: Fabric4, Refreshed: 8/16/23, 5:51:27 AM, Next refresh: N/A); Lakehouse (Owner: Hien Tran, Refreshed: —, Next refresh: —); Dataset (default) (Owner: Fabric4, Refreshed: 8/19/23, 8:53:09 PM, Next refresh: N/A); SQL endpoint (Owner: Fabric4, Refreshed: —, Next refresh: N/A); Lakehouse (Owner: Hien Tran, Refreshed: —, Next refresh: —); and a partially visible entry starting with "Lakehouse1".

- From Workspace screen > Click New > Select “Warehouse..”

The screenshot shows a "New warehouse" dialog box. It has a "Name" field containing "Warehouse01", a "Create" button in a green box, and a "Cancel" button. There is also a close button "X" in the top right corner.

- Input Warehouse name

The screenshot shows the Fabric Data Platform interface. On the left, there's a sidebar with various icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric (which is highlighted with a yellow box), and Notebook. Below the sidebar, there are tabs for Data, Query, and Model. The main area has a search bar at the top. Below the search bar, a message says "A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)". The Explorer sidebar on the left shows a tree structure for "Warehouse01": Warehouses, Schemas (dbo, guest), Security (DB roles Built-in: db_accessadmin, db_backupoperator, db_datareader, db_datawriter, db_ddladmin, db_denydatareader, db_denydatawriter, db_owner, db_securityadmin), and DB roles (Custom). To the right, there's a "Build a warehouse" section with four cards: "Create tables with T-SQL" (grid icon), "Get data with new Dataflow Gen2" (refresh icon), "Get data with new data pipeline" (play icon), and "Start with sample data" (sample icon).

2.5 ElephantSQL

Create trial account in Elephant SQL for testing loading data from external database

<https://www.elephantsql.com/plans.html>



2.5.1 Create sales table in SQL

SQL_Script

https://github.com/Huong2709/fabric/blob/main/sql_queries.txt

```
sql_queries_2          sql_queries      +  
File Edit View  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (21156, 14, '10/2/2020', '10/11/2020', 1162, 69, 1, 65, 52, 32);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (21320, 32, '10/2/2020', '10/10/2020', 1879, 62, 2, 40, 36, 13);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (22561, 26, '10/2/2020', '10/3/2020', 2954, 32, 2, 83, 66, 29);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (23121, 53, '10/2/2020', '10/10/2020', 2025, 93, 2, 91, 77, 25);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (23186, 20, '10/2/2020', '10/10/2020', 882, 27, 1, 47, 39, 1);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (25941, 40, '10/2/2020', '10/4/2020', 3869, 77, 1, 40, 38, 13);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (26619, 29, '10/2/2020', '10/7/2020', 1070, 69, 2, 91, 77, 2);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (29171, 49, '10/2/2020', '10/3/2020', 1763, 97, 2, 97, 67, 44);  
  
    INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty, Price, Cost, DiscountPercent)  
    VALUES (29189, 17, '10/2/2020', '10/5/2020', 5083, 63, 2, 47, 39, 45);
```

The screenshot shows the ElephantSQL SQL Browser interface. The left sidebar contains navigation links: DETAILS, ALARMS, BROWSER (selected), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, and FIREWALL. The top right corner shows the user 'sybspbgi' and a Fabric dropdown menu. The main area is titled 'SQL Browser' and displays the following SQL code:

```
--SQL Query to Create Table:  
  
CREATE TABLE sales (  
    OrderNo INT PRIMARY KEY,  
    ItemID INT,  
    SalesDate DATE,  
    DeliveryDate DATE,  
    CustomerId INT,  
    CityId INT,  
    Qty FLOAT,  
    Price FLOAT,  
    Cost FLOAT,  
    DiscountPercent FLOAT  
);  
  
--SQL Queries to Insert Data:  
  
INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty,  
Price, Cost, DiscountPercent)  
VALUES (20277, 37, '10/3/2020', '10/6/2020', 2691, 26, 1, 233, 172, 12);  
  
INSERT INTO sales (OrderNo, ItemID, SalesDate, DeliveryDate, CustomerId, CityId, Qty,  
Price, Cost, DiscountPercent)
```

At the bottom, there are buttons for 'Table queries ▾', 'Previous queries ▾', and 'Execute ▶'.

- Navigate to “Browser” input script for create your first table to test

2.5.2 Create sales_fact from sales

The screenshot shows the ElephantSQL interface with the 'BROWSER' tab selected. The main area displays a SQL query:

```
-- SQL query 1
select *
into sales_fact
from sales
where salesdate <= '2018-12-01'
;
select count(*)
from sales_fact
;
```

Below the query editor, there are three buttons: 'Table queries ▾', 'Previous queries ▾', and a blue 'Execute' button with a play icon. A message 'No rows returned' is displayed at the bottom.

-- SQL query 1

```
select *
into sales_fact
from sales
where salesdate <= '2018-12-01'
;
select count(*)
from sales_fact
;
```



DETAILS

ALARMS

BROWSER

STATS

SLOW QUERIES

BACKUP

LOG

METRICS

ADMIN

INTEGRATIONS

FIREWALL

MAINTENANCE

SQL Browser

```
select count(*)  
from sales_fact
```

Notifications

Dismiss all

success

Query completed



Table queries ▾

Previous queries ▾

Execute ►

count

2873

```
select count(*)
```

```
from sales_fact
```

2.6 Create pipeline in warehouse

Warehouse01 | No label

Search

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

Warehouse 01

Home

Get data

New SQL query

New visual query

New report

New measure

A default warehouse has been created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

New Dataflow Gen2

New Data pipeline

Explorer

+ Warehouses

Warehouse01

Schemas

dbo

guest

INFORMATION_SCHEMA

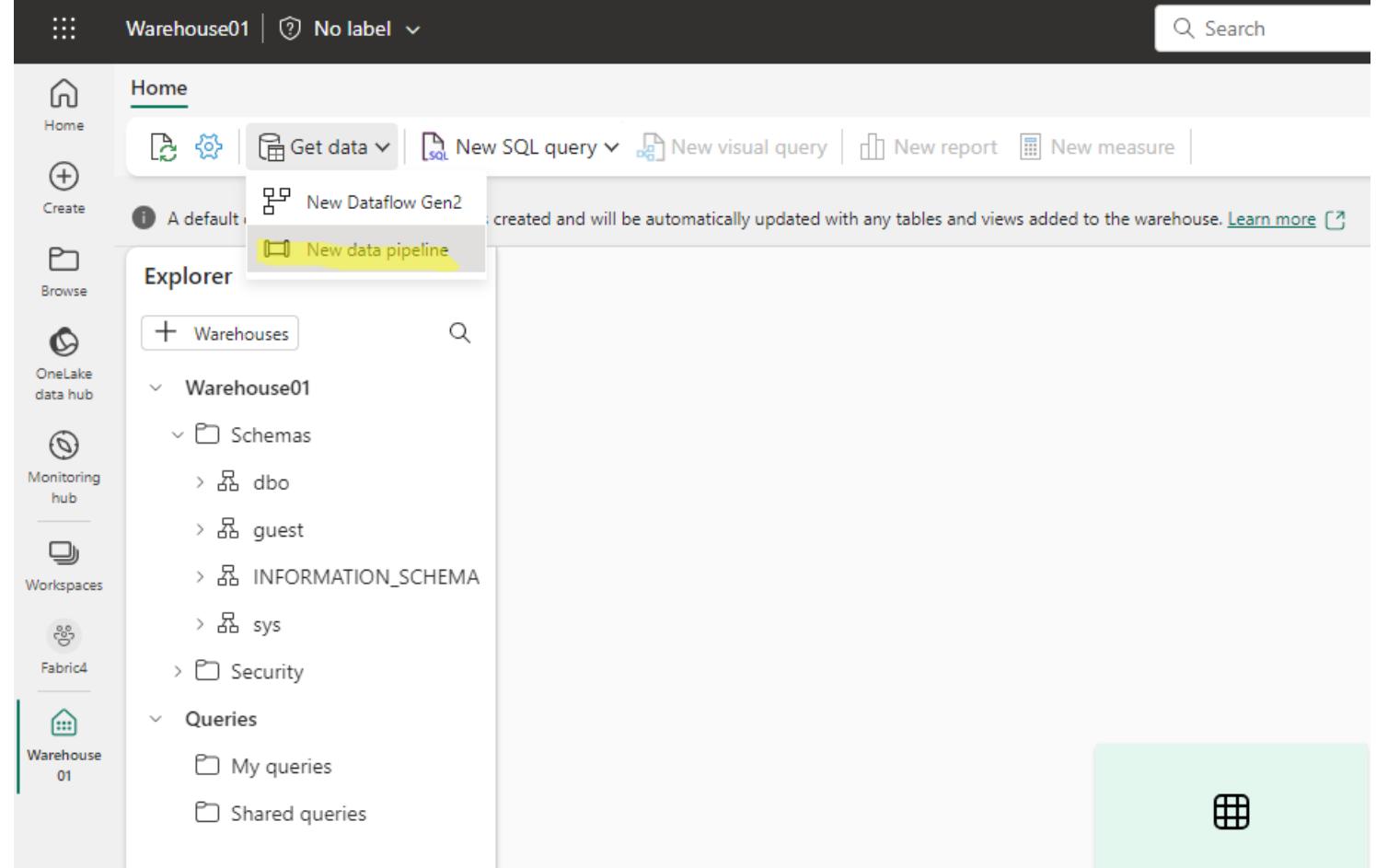
sys

Security

Queries

My queries

Shared queries



- From warehouse > click Get data > select New data pipeline

New pipeline

X

Name

pipeline01- to wh01

Create

Cancel

- Input New pipeline

pipeline... | ? No label

Search

Trial:
59 days left

Home Activities Run View

Copy data into Data warehouse

Choose data source
Select a connector. Then enter the connection information.

Connect to data source

Choose data destination

Connect to data destination

Settings

Review + save

Build your data ingestion task to move objects from a data source to a data destination. [Learn more](#)

Sample data

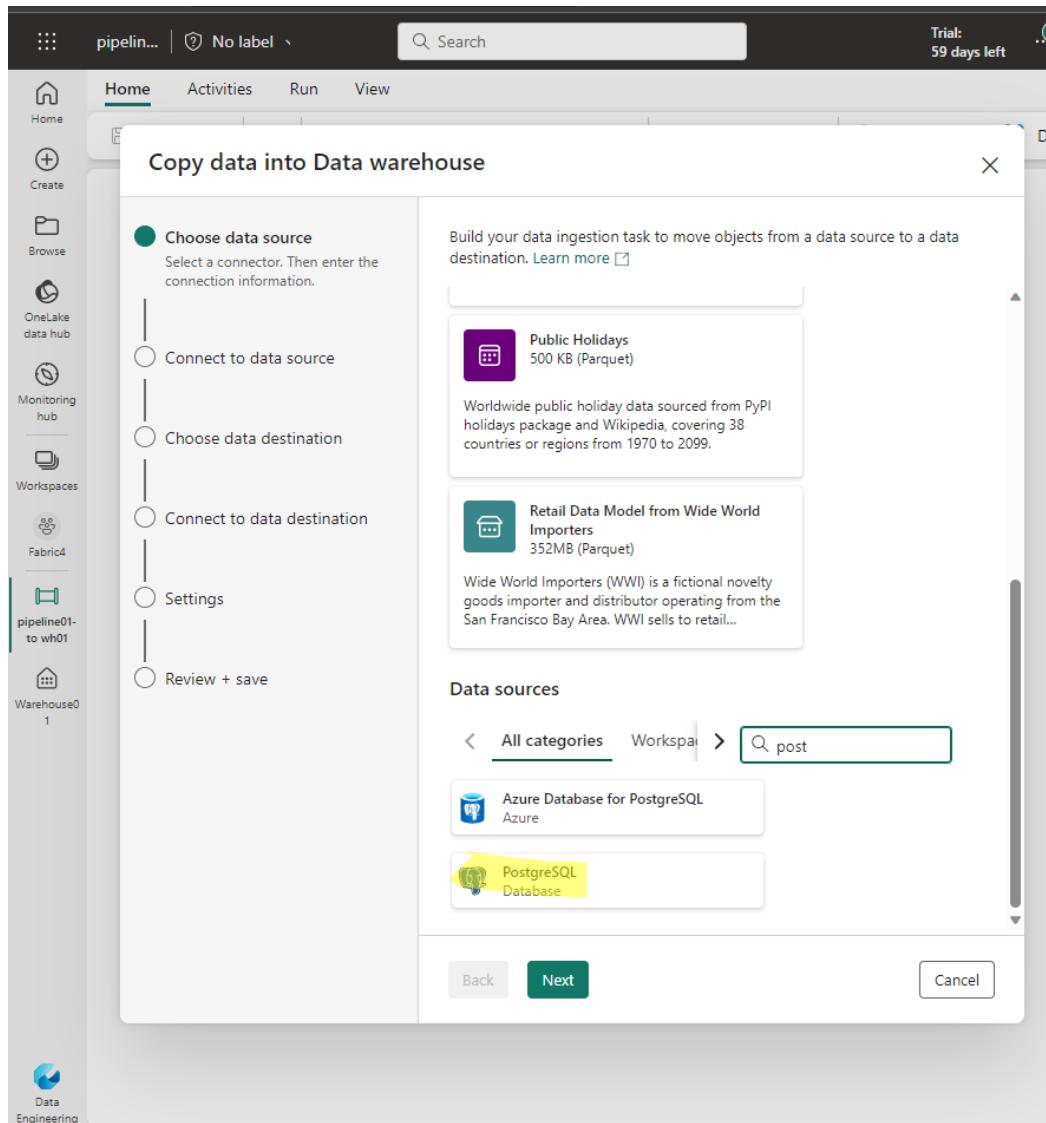
 COVID-19 Data Lake
Varied per format (CSV, JSON, JSON Lines, Parquet)
The COVID-19 Data Lake contains COVID-19 related datasets from various sources. It covers testing and patient outcome tracking data, social distancing...

 NYC Taxi - Green
2 GB (Parquet)
The green taxi trip records include fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, trip distances, itemized fares,...

 Diabetes
14 KB (Parquet)
The Diabetes dataset has 442 samples with 10 features, making it ideal for getting started with machine learning algorithms.

 Public Holidays
500 KB (Parquet)

Back Next Cancel



- Select PostgreSQL data source

The screenshot shows the 'Copy data into Data warehouse' wizard in Databricks. The 'Create new connection' option is selected. In the 'Connection settings' section, the 'Server' field is set to 'ro...' and the 'Database' field is set to 'st...'. In the 'Connection credentials' section, the 'Connection name' is 'ro...', 'Authentication kind' is 'Basic', 'Username' is '...', and 'Password' is '...'. The 'Use encrypted connection' checkbox is checked. To the right, a modal for 'ElephantSQL' displays the details of a database instance, including its URL (post...), current database size (6 MB), and max database size (20 MB). The modal also lists various monitoring and management features like alarms, browser stats, slow queries, backups, logs, metrics, and integrations.

- Input connection details
- Click Next

Copy data

Select a table

Existing tables: public.sales_fact

Use query:

Search: sale

Tables:

- publicsales_fact
- publicsales_hien

Preview data: public.sales_fact

| orderno | itemid | salesdate | deliverydate | customerid | cityid | qty | price | cost | discountpercent |
|---------|--------|---------------------|---------------------|------------|--------|-----|-------|------|-----------------|
| 1210 | 5 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 246 | 67 | 2 | 92 | 64 | 17 |
| 2130 | 8 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1457 | 96 | 1 | 52 | 36 | 11 |
| 2353 | 26 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1327 | 40 | 1 | 167 | 115 | 4 |
| 2539 | 23 | 2018-12-01T00:00:00 | 2018-12-11T00:00:00 | 3569 | 59 | 1 | 83 | 66 | 2 |
| 3367 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 2470 | 51 | 1 | 67 | 53 | 34 |
| 3416 | 25 | 2018-12-01T00:00:00 | 2018-12-01T00:00:00 | 1907 | 84 | 2 | 753 | 376 | 25 |
| 3561 | 26 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 105 | 30 | 2 | 1333 | 666 | 34 |
| 3566 | 29 | 2018-12-01T00:00:00 | 2018-12-08T00:00:00 | 2711 | 34 | 1 | 167 | 115 | 28 |
| 3717 | 5 | 2018-12-01T00:00:00 | 2018-12-10T00:00:00 | 3763 | 69 | 2 | 753 | 376 | 32 |
| 4117 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 86 | 86 | 2 | 91 | 77 | 26 |

Back Next Cancel

- From connect to data source step > Select Public.sales_fact > click Next

Copy data

Data destinations

All categories Workspace Azure Database Generic protocol Services and apps

Search: Q

Destinations:

- Data Warehouse Workspace
- KQL Database Workspace
- Lakehouse Workspace

Back Next Cancel

- From Choose data destination > select Workspace > Click Data warehouse > click Next



Home

Home

Activities

Run

View



Create



Browse



OneLake

data hub



Monitoring

hub



Workspaces



Fabric4

pipeline01-
to wh01

Warehouse0

1



Data

Engineering

Copy data

Choose data source



Data Warehouse
[Learn more](#)

Data Warehouse *

Warehouse01

Refresh

Connect to data source

Choose data destination

Define the data store as destination.

Connect to data destination

Settings

Review + save

Back

Next

- Select Data warehouse

pipeline01- to wh01 | No label

Search

Trial: 59 days left

Home Activities Run View

Copy data

Choose data source

Connect to data source

Choose data destination

Connect to data destination
Select and map to table.

Settings

Review + save

Load settings

Load to new table Load to existing table

Destination table name *

Column mappings

New mapping Reset Delete

| Source | Type | Destination | Type |
|-----------------|----------|-----------------|----------|
| orderno | Int32 | orderno | Int32 |
| itemid | Int32 | itemid | Int32 |
| salesdate | DateTime | salesdate | DateTime |
| deliverydate | DateTime | deliverydate | DateTime |
| customerid | Int32 | customerid | Int32 |
| cityid | Int32 | cityid | Int32 |
| qty | Double | qty | Double |
| price | Double | price | Double |
| cost | Double | cost | Double |
| discountpercent | Double | discountpercent | Double |

Back Next Cancel

pipeline01- to wh01

Warehouse0 1

Data Engineering

- Select “Load to new table”
- Destination table name: Provide table name in Fabric
- Select Source to load
- Click Next

pipeline01-to wh01 | No label | 59 days left | L | X

Home Activities Run View

Copy data

Choose data source

Connect to data source

Choose data destination

Connect to data destination

Settings

Review + save

Enable staging

Copy command settings

Default values

Back Next Cancel

pipe... pipeline01-to wh01

W... Warehouse0 1

Data Engineering

- Click Next

pipeline01- to wh01 | No label

Search

Trial: 59 days left

Home Activities Run View

Copy data

Choose data source

Connect to data source

Choose data destination

Connect to data destination

Settings

Review + save

Confirm Copy activity summary

Copy Summary

PostgreSQL → Staging → Destination

| Source | Staging | Destination |
|-----------------|--|-----------------|
| Connection name | rosi e.d b.el eph ants ql.c om; sybs pbgi htra n (2) | Connection name |
| Table name | publ ic.sa les_f act | Table name |

Options

Start data transfer immediately

Back Save + Run Cancel

The screenshot shows the 'Copy data' step in a pipeline creation wizard. The left sidebar lists various components: pipeline01-to wh01, OneLake data hub, Monitoring hub, Workspaces, Fabric4, pipeline01-to wh01, Warehouse0 1, and Data Engineering. The main panel displays the 'Copy data' configuration with a flow diagram showing data moving from PostgreSQL to Staging to Destination. The 'Save + Run' button is highlighted.

- Click "Save + Run"

Pipeline

Copy data step is automatically creation after completion pipeline creation step in the above

pipeline... | No label | Search | Trial:
59 days left

Home Activities Run View

Home Create Browse OneLake Monitoring hub Workspaces Fabric4 pipeline01-to wh01 Warehouse0 1 Data Engineering

Copy data

Copy_twd

✓ ↻ ✓ ✎ ↪

✗ ↵ ↷ ↴ ↸ ↹

General Source Destination Mapping Settings

Name * Copy_twd Learn more ↗

Description

Timeout ⓘ 0.12:00:00

Retry ⓘ 0

> Advanced



Home

Home



Get data ▾



New SQL query ▾



New visual query



Create



Browse

OneLake
data hubMonitoring
hub

Workspaces



Fabric4

Warehouse
01pipeline01-
to wh01Data
Engineering

Explorer

+ Warehouses



Warehouse01

Schemas

dbo

Tables

sales

Views

Functions

Stored Procedures

guest

INFORMATION_SCHEMA

sys

Security

Queries

My queries

SQL query 1

Shared queries

SQL query 1

Run Save as view

```
1 select count(*)  
2 from sales
```

Messages

Results

Save as table

...

123 untitled1

1

2873

Succeeded (1 sec 613 ms)

Data

Query

Model

2.7 Create pipeline from Workspace

The screenshot shows the Fabric4 workspace interface. On the left, there's a sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, and Data Engineering. The 'Workspaces' icon is highlighted with a yellow box. The main area shows a list of items under 'Fabric4'. A yellow box highlights the 'Data pipeline (Preview)' option in the 'New' dropdown menu. A tooltip for 'Data pipeline (Preview)' says 'Ingest data at scale and schedule data workflows.' Below the dropdown, there's a table listing various items with columns for Type, Owner, Refreshed, and Next r. The table includes entries like Dataset, Notebook, Experiment, Model, Notebook, Experiment, Report, Data pipeline, Report, Report, Dashboard, Dataset (default), and Warehouse.

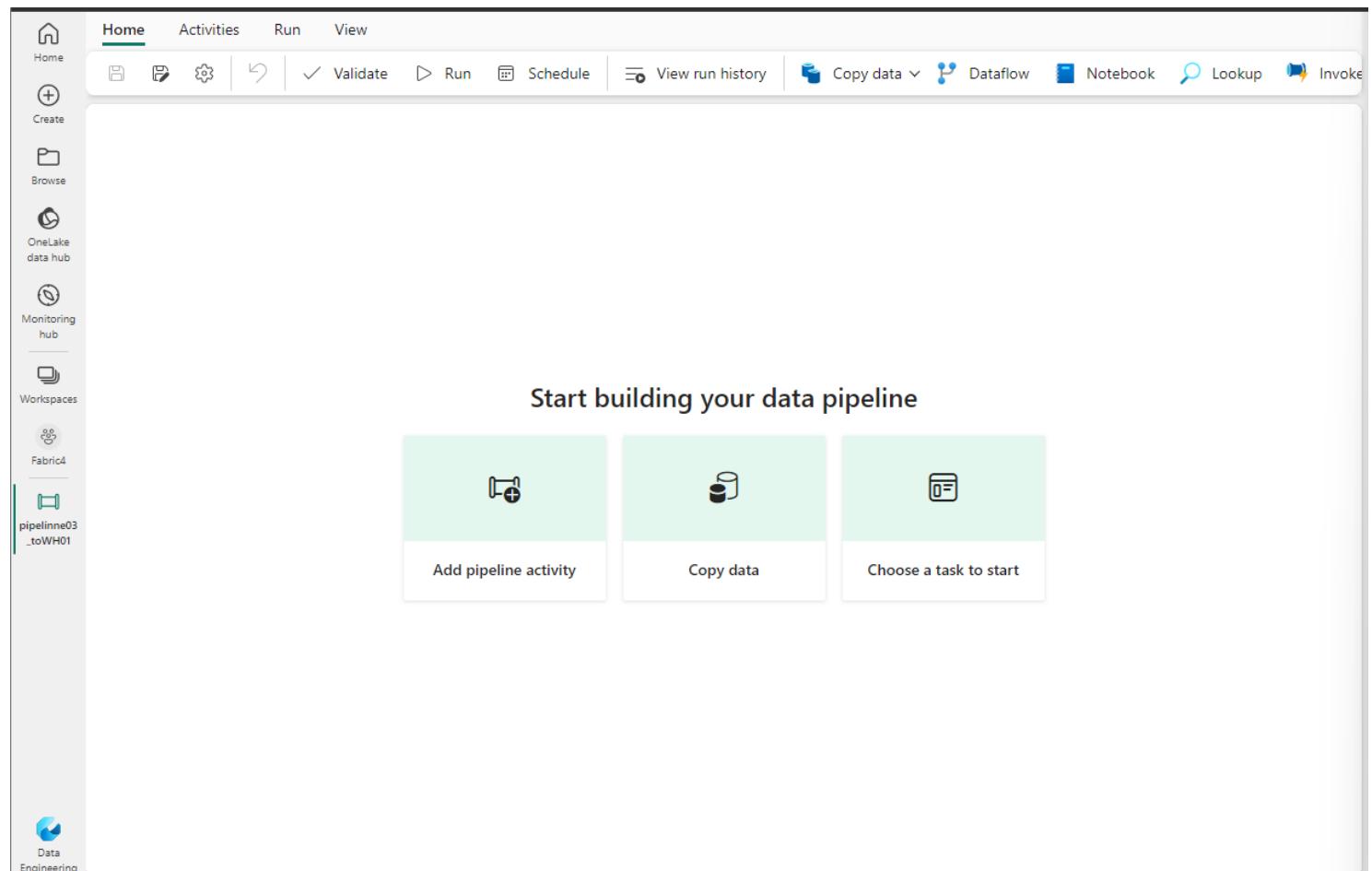
| Type | Owner | Refreshed | Next r |
|-------------------|-----------|---------------------|--------|
| Dataset | Fabric4 | 8/20/23, 0:10:23 AM | N/A |
| Notebook | Hien Tran | — | — |
| Notebook | Hien Tran | — | — |
| Notebook | Hien Tran | — | — |
| Experiment | Hien Tran | — | — |
| Notebook | Hien Tran | — | — |
| Data pipeline | Hien Tran | — | — |
| Report | Fabric4 | 8/15/23, 7:07:14 PM | — |
| Report | Fabric4 | 8/26/23, 6:16:23 AM | — |
| Dashboard | Fabric4 | — | — |
| Dataset (default) | Fabric4 | 8/25/23, 6:25:36 AM | N/A |
| Warehouse | Hien Tran | 8/26/23, 6:43:21 AM | N/A |

New pipeline

Name

Create

Cancel



Activity 1: Lookup

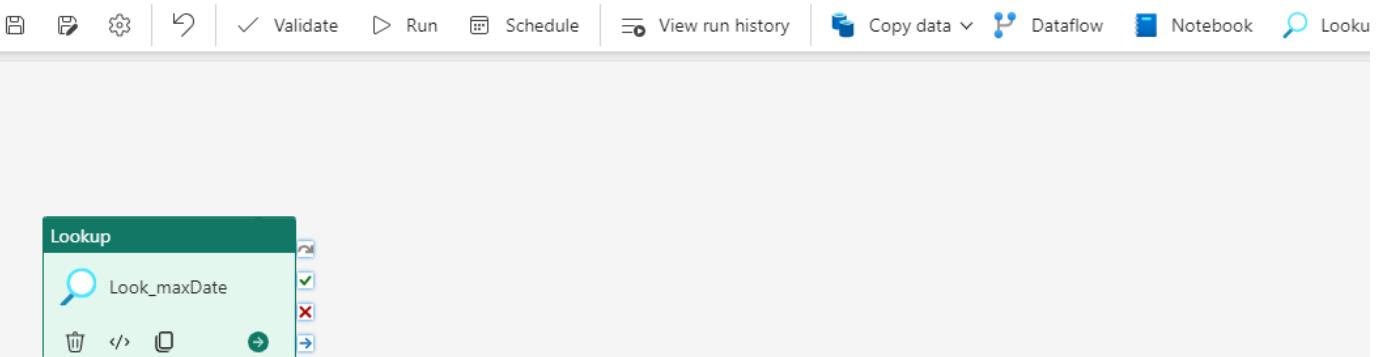
The screenshot shows the Azure Data Factory pipeline editor for the "pipelinne03_toWH01" pipeline. The pipeline name is displayed at the top. The "Lookup" activity is selected, shown in a green box with the name "Lookup1". The pipeline editor interface includes tabs for Home, Activities, Run, View, and a toolbar with icons for File, Copy, Paste, Settings, Validate, Run, Schedule, View run history, Copy data, Dataflow, Notebook, and a yellow-highlighted "Lookup" icon.

General tab settings:

- Name: Learn more [🔗](#)
- Description:
- Timeout:
- Retry:
- [Advanced](#)

Name the activity

Home Activities Run View

pipeline03
_toWH01General Settings ¹Name * Learn more Description Timeout ⓘ Retry ⓘ [Advanced](#)Data
Engineering

pipeline03_toWH01 | No label

Search

Trial:
59 days left

Home Activities Run View

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

pipelinne03_toWH01

General Settings

Data store type Workspace External

Workspace data store type

Data Warehouse Refresh Open

Use query Table Query Stored procedure

Query *
select max(salesdate) maxDate from sales

Edit Preview data

First row only

Advanced

Data Engineering

Preview data

| | maxDate |
|---|---------------------|
| 1 | 2018-12-01T00:00:00 |

Save activity



Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow Notebook Lookup Invoke

Home Activities Run View

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

pipelinne03_toWH01

Lookup

Look_maxDate

General Settings

Data store type Workspace External

Workspace data store type

Data Warehouse Refresh Open

Use query Table Query Stored procedure

Query * Edit Preview data

First row only

> Advanced

Data Engineering

Run

pipeline03_toWH01 | No label

Search

Trial:
59 days left

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow Notebook Lookup

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

General Settings

Data store type Workspace External

Workspace data store type

Data Warehouse Refresh Open

Use query Table Query Stored procedure

Query * `select max(salesdate) maxDate from sales` Edit Preview data

First row only

> Advanced

Data Engineering

The screenshot shows the Azure Data Factory pipeline editor interface. On the left, there's a sidebar with icons for Home, Activities, Run, View, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and a selected pipeline named 'pipeline03_toWH01'. The main area displays a pipeline component named 'Lookup' with a single activity named 'Look_maxDate'. Below this, the 'Output' tab is selected, showing a table of pipeline run details. The table includes columns for Activity name, Activity status, Activity type, Run start, Duration, and Activity run ID. The 'Look_maxDate' activity is listed with a status of 'Succeeded'.

| Activity name | Activity status | Activity type | Run start | Duration | Activity run ID |
|---------------|--|---------------|-----------------------|----------|-----------------|
| Look_maxDate | Succeeded | Lookup | 8/26/2023, 6:53:18 AM | 4s | c03e... |

2.7.1 Preparation for creating a pipeline with many steps

Open Warehouse

The screenshot shows the Azure Data Lake Gen2 Explorer interface. The sidebar on the left includes icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, Fabric4, and a selected warehouse named 'Warehouse01'. The main area features a toolbar with 'Get data', 'New SQL query' (which is highlighted with a yellow box), 'New visual query', and 'New report'. Below the toolbar is an 'Explorer' pane showing a tree structure for 'Warehouse01' with nodes for Warehouses, Schemas, Security, Queries, My queries, and Shared queries.

- Click New Query

Create GrossByDate table

The screenshot shows the Azure Data Studio interface. On the left, the sidebar lists various workspaces and databases. The 'Warehouse 01' section is selected. In the center, the 'Query' tab is active, showing two tabs: 'SQL query 1' and 'SQL query 2'. 'SQL query 2' is highlighted with a yellow box. The code in 'SQL query 2' is:

```
-- Create aggregation table in WH
DROP TABLE IF EXISTS dbo.GrossByDate;
CREATE TABLE dbo.GrossByDate
(
    salesdate datetime2(6),
    gross FLOAT
)
```

Below the code, the 'Messages' pane shows the execution details:

- 6:34:55 AM Started running query at line 1
- Statement ID: {2135D804-08D7-4E60-B2C7-31430E563EA8}
- Msg 24528, Level 0, State 1
- Statement ID: {F0A54A71-E4A7-478B-BEC2-A33EBA7B6D71}
- Msg 24528, Level 0, State 1
- 6:34:57 AM Total execution time: 00:00:02.889

At the bottom, a green status bar indicates 'Succeeded (2 sec 889 ms)'.

- Input query

```
-- Create aggregation table in WH02
```

```
DROP TABLE IF EXISTS dbo.GrossByDate
```

```
;
```

```
CREATE TABLE dbo.GrossByDate
```

```
(
```

```
    salesdate datetime2(6),
```

```
    gross FLOAT
```

```
)
```

- Run

Insert data

The screenshot shows the Azure Data Studio interface. At the top, there's a navigation bar with 'LakeHouse10 (1)' and a 'No label' dropdown. A search bar is on the right. On the far right, there's a 'Trial: 59 days left' message, a bell icon with '1' notification, a gear icon, and a question mark.

The main area has a sidebar on the left with icons for Home, Create, Browse, Workspaces, Fabric4, Notebook 3, Notebook 2, LakeHouse1, Warehouse 01, pipeline01-to wh01, and three ellipsis items. Below the sidebar is a 'Data Engineering' section with 'Data', 'Query' (which is selected), and 'Model' tabs.

The central workspace is divided into several sections:

- Explorer:** Shows a tree view of databases, schemas, tables, and views. Under 'Warehouse01', 'dbo' contains 'GrossByDate' and 'sales' tables, and 'Views'.
- SQL query 3:** The active tab, containing the following SQL code:

```
1  insert into dbo.GrossByDate
2  select a.* from
3  (select salesdate ,
4    SUM(qty * price) as gross
5    from dbo.sales
6    group by salesdate
7  ) a
```
- Messages:** A log of events:
 - Started running query at line 1
 - Statement ID: {38D9432F-9B4B-4F6E-8395-0ED6A478451D} | Query hash: 0xA214FFE48E8E8B4C | [request ID: {B8C2C6EF-FF3A-4F28-B9FA-15A8F3EC04AO}]
Msg 15806, Level 0, State 1
(50 records affected)
 - Total execution time: 00:00:12.089
- Status:** Shows a green success icon and 'Succeeded (12 sec 89 ms)'.

insert into dbo.GrossByDate

select a.* from

(select salesdate ,

 SUM(qty * price) as gross

 from dbo.sales

 group by salesdate

) a

LakeHouse10 (1) | No label

Search

Trial:
59 days left

Home Reporting Table tools

Get data New SQL query New visual query New report New measure

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Manage default Power BI data

Explorer

+ Warehouses

Warehouse01

Schemas

dbo

Tables

GrossByDate

sales

cityid

cost

customerid

deliverydate

discountpercent

itemid

orderno

price

qty

salesdate

Views

Data preview

Showing 50 rows

| | salesdate | gross |
|----|----------------------------|-------|
| 1 | 2018-11-12 00:00:00.000000 | 41134 |
| 2 | 2018-10-29 00:00:00.000000 | 36861 |
| 3 | 2018-10-22 00:00:00.000000 | 57721 |
| 4 | 2018-11-30 00:00:00.000000 | 25402 |
| 5 | 2018-11-05 00:00:00.000000 | 44811 |
| 6 | 2018-10-15 00:00:00.000000 | 46326 |
| 7 | 2018-11-09 00:00:00.000000 | 34753 |
| 8 | 2018-11-02 00:00:00.000000 | 48878 |
| 9 | 2018-10-19 00:00:00.000000 | 36748 |
| 10 | 2018-11-23 00:00:00.000000 | 36358 |
| 11 | 2018-11-19 00:00:00.000000 | 47462 |
| 12 | 2018-11-15 00:00:00.000000 | 44640 |
| 13 | 2018-10-25 00:00:00.000000 | 49725 |
| 14 | 2018-11-22 00:00:00.000000 | 31760 |
| 15 | 2018-11-13 00:00:00.000000 | 30534 |
| 16 | 2018-12-01 00:00:00.000000 | 48945 |
| 17 | 2018-11-27 00:00:00.000000 | 51167 |
| 18 | 2018-11-06 00:00:00.000000 | 46915 |
| 19 | 2018-10-16 00:00:00.000000 | 36786 |
| 20 | 2018-11-26 00:00:00.000000 | 33242 |
| 21 | 2018-11-10 00:00:00.000000 | 49355 |
| 22 | 2018-11-01 00:00:00.000000 | 45582 |

Succeeded (1 sec 365 ms)

Columns: 2

Data Query Model

LakeHouse10 (1) | No label | Search | Trial: 59 days left | Home | Create | Get data | New SQL query | New visual query | Manage default Power BI dataset

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Explorer

- + Warehouses
- Warehouse01
 - Schemas
 - dbo
 - GrossByDate
 - gross
 - salesdate
 - sales
 - cityid
 - cost
 - customerid
 - deliverydate
 - discountpercent
 - itemid
 - orderno
 - price
 - qty

SQL query 1 | SQL query 2 | SQL query 3 | SQL query 4

Run | Save as view

```
1 SELECT *
2 FROM GrossByDate
3 ORDER BY salesdate DESC
```

Messages | Results | Save as table | Download Excel file | Visualize results

Search

| | salesdate | gross |
|----|----------------------------|-------|
| 1 | 2018-12-01 00:00:00.000000 | 48945 |
| 2 | 2018-11-30 00:00:00.000000 | 25402 |
| 3 | 2018-11-29 00:00:00.000000 | 40913 |
| 4 | 2018-11-28 00:00:00.000000 | 35512 |
| 5 | 2018-11-27 00:00:00.000000 | 51167 |
| 6 | 2018-11-26 00:00:00.000000 | 33242 |
| 7 | 2018-11-25 00:00:00.000000 | 34725 |
| 8 | 2018-11-24 00:00:00.000000 | 34120 |
| 9 | 2018-11-23 00:00:00.000000 | 36358 |
| 10 | 2018-11-22 00:00:00.000000 | 31760 |
| 11 | 2018-11-21 00:00:00.000000 | 42232 |
| 12 | 2018-11-20 00:00:00.000000 | 30329 |
| 13 | 2018-11-19 00:00:00.000000 | 47462 |

Succeeded (1 sec 533 ms)

Columns: 2 Rows: 13

Data | Query | Model

Create storeprocedure

The screenshot shows the Power BI Data Explorer interface. On the left, the Explorer pane displays a tree structure of databases, schemas, tables, views, functions, and stored procedures. The 'Stored Procedures' node under 'Warehouse01' is expanded, showing a single entry: 'Proc_GrossByDate'. The main workspace contains five tabs labeled 'SQL query 1' through 'SQL query 5'. The 'SQL query 1' tab contains the following T-SQL code:

```
1 insert into dbo.GrossByDate
2 select a.* from
3 (select salesdate ,
4 SUM(qty * price) as gross
5 from dbo.sales
6 group by salesdate
7 ) a
```

The 'Messages' pane at the bottom right shows the execution log:

- Started running query at line 1
- Statement ID: {A522BC24-B461-4569-95A4-3A3B8F94511A} | Query hash: 0xA214FFE48E8E8B4C | Distributed request ID: {97D8F885-3092-4363-81D3-59E86533CA7A}
- Msg 15806, Level 0, State 1
- (50 records affected)
- Total execution time: 00:00:07.824
- Succeeded (7 sec 824 ms)

The status bar at the bottom indicates 'Columns: 0 Row: 0'.

This screenshot shows the same Power BI Data Explorer interface, but with a different set of objects in the Explorer pane. The 'Stored Procedures' node under 'Warehouse01' is highlighted with a yellow box, and the specific procedure 'Proc_GrossByDate' is also highlighted with a yellow box. The 'SQL query 1' tab contains the following T-SQL code, which includes syntax highlighting for identifiers like 'Name', 'Type', and 'Scope':

```
1 -- Create procedure Proc_GrossByDate
2 CREATE PROCEDURE dbo.Proc_GrossByDate
3 AS
4 BEGIN
5
6 insert into dbo.GrossByDate
7 select a.* from
8 (select salesdate,
9 SUM(qty * price) as gross
10 from dbo.sales
11 where salesdate > @start_date and salesdate < @end_date
12 group by salesdate
13 ) a
14
15
16 end
17
18 GO
```

The 'Messages' pane shows the execution results:

- Started running query at line 1
- Commands completed successfully.
- Total execution time: 00:00:01.907
- Succeeded (1 sec 907 ms)

The status bar at the bottom indicates 'Columns: 0 Rows: 0'.

```
-- Create procedure Proc_GrossByDate
```

```
CREATE PROCEDURE dbo.Proc_GrossByDate
```

```
AS
```

```
BEGIN
```

```
insert into dbo.GrossByDate
```

```
select a.* from
```

```
(select salesdate ,
```

```
    SUM(qty * price) as gross
```

```
    from dbo.sales
```

```
    where salesdate > (select max(salesdate) from dbo.GrossByDate)
```

```
    group by salesdate
```

```
) a
```

```
end
```

```
GO
```

2.7.2 Set Pipeline

Step 1: Lookup

The screenshot shows the Azure Data Factory pipeline editor interface. On the left, there's a sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, and Fabric4. The main area shows a pipeline named 'pipeline03_toWH01'. A 'Lookup' activity is selected, with its name 'Look_maxDate' highlighted. Below the activity, there are tabs for Parameters, Variables, Settings, and Output. The 'Settings' tab is active, showing fields for Concurrency (empty), General (empty), and Advanced (empty). The 'Concurrency' field has a yellow highlight.

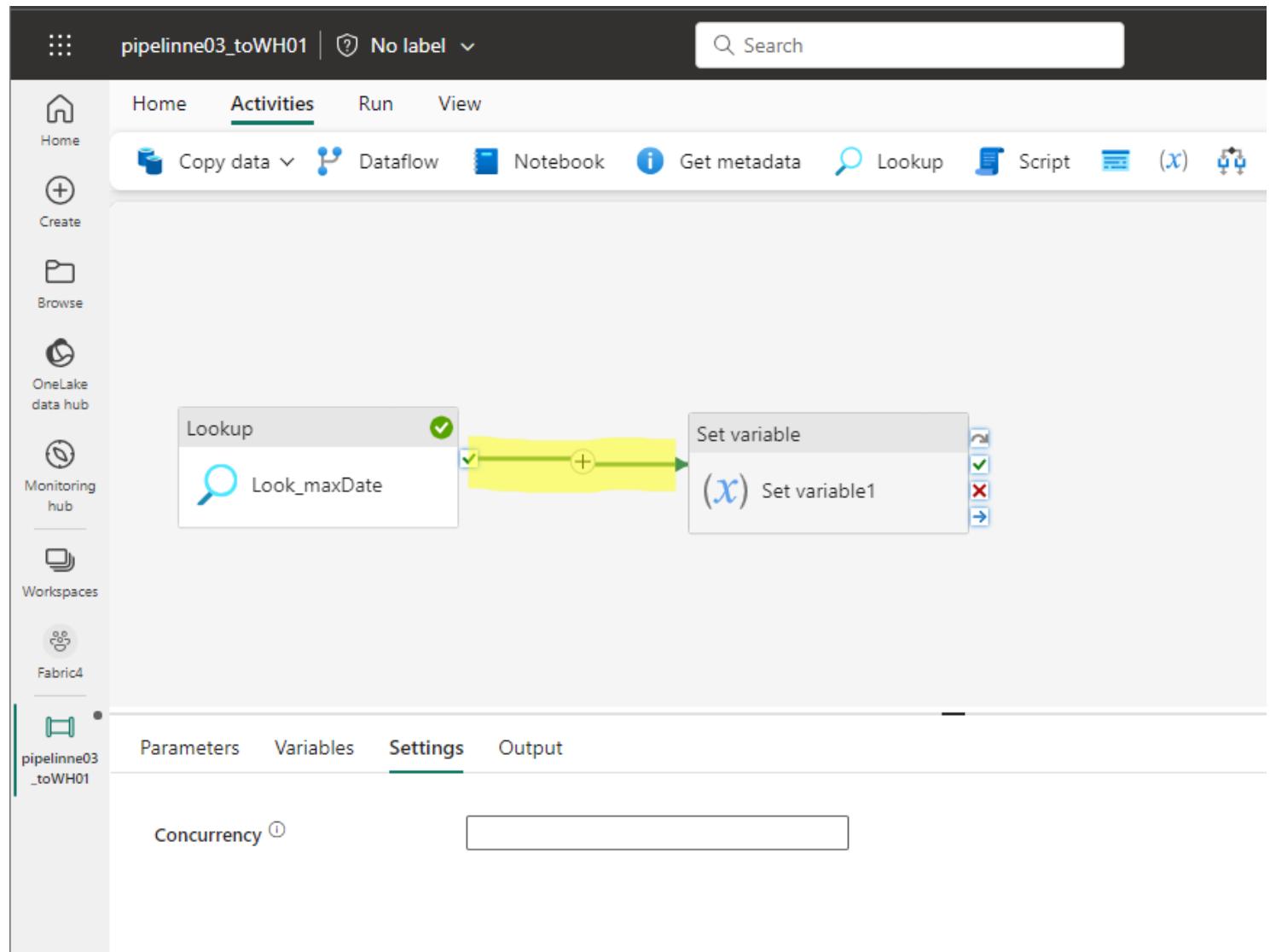
This screenshot shows the 'General' settings for the 'Look_maxDate' Lookup activity. It includes fields for Name (Look_maxDate), Description (empty), Timeout (0.12:00:00), and Retry (0). There's also an 'Advanced' link. The 'Name' field has a yellow highlight.

This screenshot shows the 'Settings' tab for the 'Look_maxDate' Lookup activity. It includes fields for Data store type (Workspace, highlighted with a yellow circle), Workspace data store type (Data Warehouse, highlighted with a yellow rectangle), Data Warehouse (Warehouse01), Refresh (button), Open (button), Use query (Query, highlighted with a yellow circle), Query (select max(salesdate) maxDate from sales, highlighted with a yellow rectangle), Edit (button), Preview data (button), First row only (checkbox checked), and Advanced (link). The 'Query' field has a yellow highlight.

```
select max(salesdate) maxDate from sales
```

After complete the step, Save > validate > run to check if any error in the setup

Step 2: Set variable



pipelinne03_toWH01 | ⓘ No label

Q Search

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script (x) ⚙️

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

pipeline03_toWH01

The screenshot shows a data flow in the Azure Data Factory pipeline editor. A 'Lookup' activity named 'Look_maxDate' is connected to a 'Set variable' activity named '(x) set_maxDate'. The 'set_maxDate' activity has a green border, indicating it is selected or highlighted. The pipeline is currently in the 'General' tab of the configuration pane.

General Settings ¹

Name * Learn more ↗

Description

> Advanced

pipeline03_toWH01 | ? No label

Search

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script

Activities

Lookup Look_maxDate

Set variable (x) set_maxDate

General Settings 1

Variable type Pipeline variable Pipeline return value

Name * + New

Value *

The screenshot shows the Azure Data Factory pipeline editor interface. At the top, there's a navigation bar with 'Home', 'Activities' (which is underlined), 'Run', and 'View'. Below the navigation bar are several icons for different activities: 'Copy data', 'Dataflow', 'Notebook', 'Get metadata', 'Lookup', 'Script', and 'Create'. On the left side, there's a sidebar with 'Home', 'Create', 'Browse', 'OneLake data hub', 'Monitoring hub', and 'Workspaces' sections. Under 'Workspaces', there are two entries: 'Fabric4' and 'pipeline03_toWH01', where the latter is selected. The main workspace area displays a pipeline diagram. It starts with a 'Lookup' activity named 'Look_maxDate', which has a green checkmark indicating it's successful. An arrow points from this activity to a 'Set variable' activity. The 'Set variable' activity is also successful, indicated by a green checkmark. The 'Set variable' activity is configured to set a pipeline variable named 'set_maxDate'. In the bottom half of the workspace, there's a 'Settings' tab with a '1' badge. Under 'Settings', there's a 'Variable type' section with two radio buttons: 'Pipeline variable' (which is selected) and 'Pipeline return value'. Below that are fields for 'Name' (with a placeholder and a '+ New' button) and 'Value' (an empty text area).

pipeinne03_toWH01 | No label

Search

Trial:
59 days left

Activities

Copy data Dataflow Notebook Get metadata

Home Create Browse OneLake data hub Monitoring hub Workspaces Fabric4

General Settings

Variable type Pipeline variable Pipeline return value

Name * maxDate

Value * @activity('look_maxDate').output.firstRow.maxDate

Add dynamic content [Alt+Shift+D]

Set variable (x) set_maxDate

Lookup Look_maxDate

Activity outputs Parameters System variables Function

Clear contents

Look_maxDate Look_maxDate activity output

Look_maxDate Look_maxDate pipeline return value

Look_maxDate first row Data of the first row

OK Cancel

```
@activity('look_maxDate').output.firstRow.maxDate
```

https://app.fabric.microsoft.com/groups/e0ba0ce0-65a9-4507-9700-d51ac507... Trial: 59 d

pipeline03_toWH01 | No label ▾ Search

Home Activities Run View

Home Create Browse OneLake Monitoring Workspaces Fabric4 pipeline03_toWH01

Validate Run Schedule View run history Copy data Dataflow

The screenshot shows a Microsoft Fabric pipeline named "pipeline03_toWH01". The pipeline consists of two main components: a "Lookup" activity named "Look_maxDate" and a "Set variable" activity. A green arrow points from the "Look_maxDate" activity to the "Set variable" activity. The "Set variable" activity is configured with the variable name "maxDate" and the value "@activity('look_maxDate').output.firs...". The pipeline is currently in the "Validate" state, indicated by a yellow highlight on the "Validate" button in the top navigation bar.

General Settings

Variable type Pipeline variable Pipeline return value

Name * maxDate [New](#)

Value @activity('look_maxDate').output.firs...

After complete the step, Save > validate > run to check if any error in the setup

pipeline03_toWH01 | No label

Search

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script (x) (y) (z) (a) (b) (c)

Home Create Browse OneLake data hub Monitoring hub Workspaces Fabric4 pipeline03_toWH01

General Settings¹

Variable type Pipeline

Name *

Lookup Look_maxDate

Set variable (x) set_maxDate

Add new variable

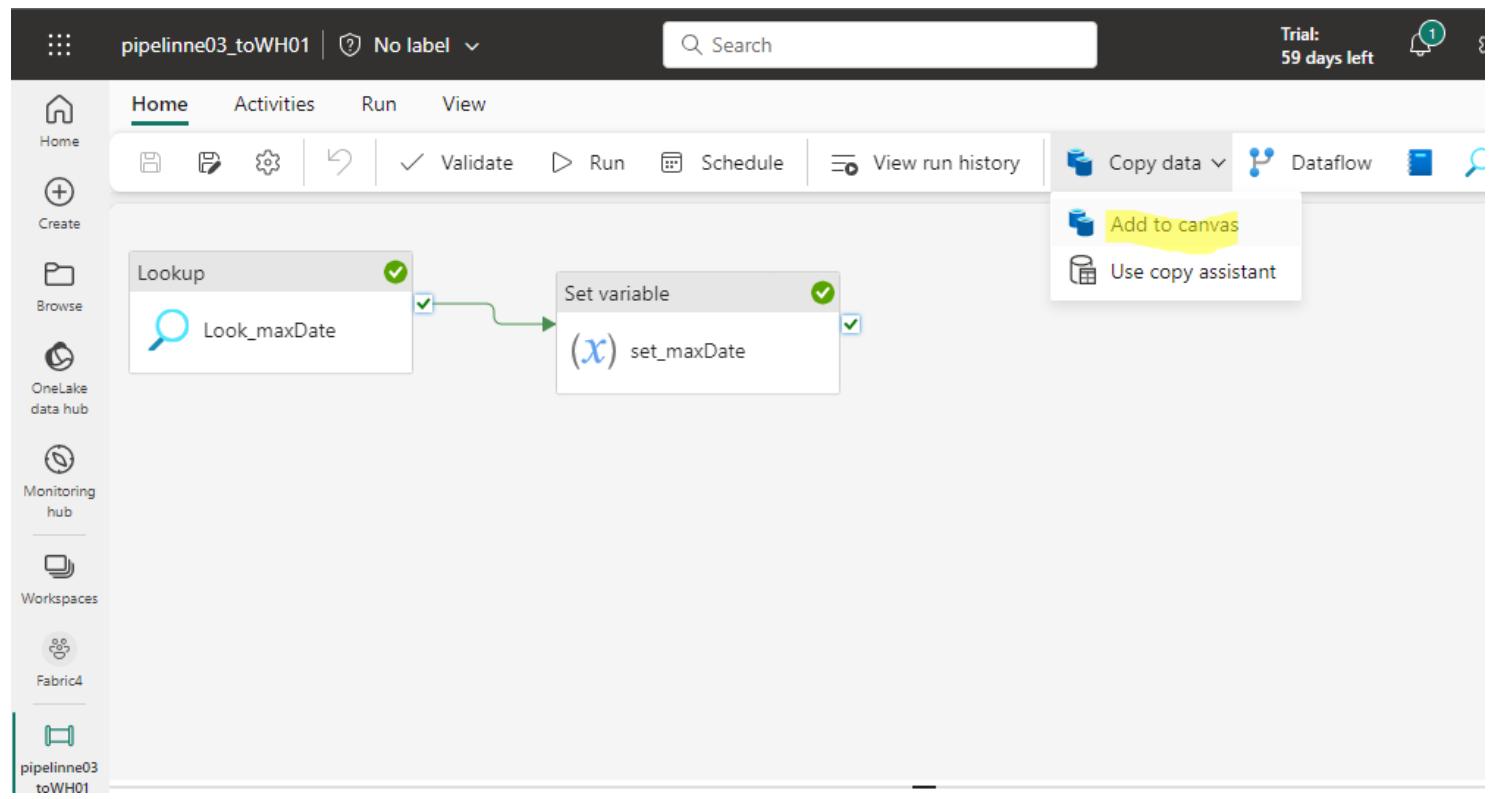
Name maxDate

Type String

Confirm Cancel

The screenshot shows the Azure Data Factory pipeline editor interface. On the left, there's a sidebar with various navigation options like Home, Activities, Run, View, and a list of workspaces. The current workspace is 'pipeline03_toWH01'. In the main area, a pipeline is being edited. A 'Set variable' activity is connected to a 'Lookup' activity. A modal dialog titled 'Add new variable' is open, prompting for a 'Name' (set to 'maxDate') and 'Type' (set to 'String'). At the bottom of the dialog are 'Confirm' and 'Cancel' buttons.

Step 3: Copy data



Screenshot of the Azure Data Factory pipeline editor showing a flow from a Lookup activity to a Set variable activity, which then triggers a Copy data activity.

```

graph LR
    L1[Lookup: Look_maxDate] --> S1[Set variable: set_maxDate]
    S1 --> C1[Copy data]
  
```

The pipeline consists of three main components:

- Lookup Activity:** Named "Look_maxDate". It has a green checkmark indicating successful validation.
- Set variable Activity:** Named "(x) set_maxDate". It has a green checkmark indicating successful validation.
- Copy data Activity:** A sub-component of the Set variable activity. It is titled "Copy data" and includes options for "Copy Data from SQL".

A green arrow connects the output of the Lookup activity to the input of the Set variable activity. Another green arrow connects the output of the Set variable activity to the input of the "Copy Data from SQL" component. A yellow arrow points to the "Copy Data from SQL" component, highlighting it.

The "Copy data" section of the pipeline configuration shows the following settings:

| Setting | Value |
|-------------|--------------------|
| Name * | Copy Data from SQL |
| Description | (Empty) |
| Timeout ⓘ | 0.12:00:00 |
| Retry ⓘ | 0 |
| > Advanced | |

pipeline03_toWH01 | No label

Search

Trial:
59 days left

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow

Lookup Look_maxDate

Preview data

| | orderno | itemid | salesdate | deliverydate | customerid | cityid | qty | price | cost | discountpercent |
|----|---------|--------|---------------------|---------------------|------------|--------|-----|-------|------|-----------------|
| 1 | 1210 | 5 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 246 | 67 | 2 | 92 | 64 | 17 |
| 2 | 2130 | 8 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1457 | 96 | 1 | 52 | 36 | 11 |
| 3 | 2353 | 26 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1327 | 40 | 1 | 167 | 115 | 4 |
| 4 | 2539 | 23 | 2018-12-01T00:00:00 | 2018-12-11T00:00:00 | 3569 | 59 | 1 | 83 | 66 | 2 |
| 5 | 3367 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 2470 | 51 | 1 | 67 | 53 | 34 |
| 6 | 3416 | 25 | 2018-12-01T00:00:00 | 2018-12-01T00:00:00 | 1907 | 84 | 2 | 753 | 376 | 25 |
| 7 | 3561 | 26 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 105 | 30 | 2 | 1333 | 666 | 34 |
| 8 | 3566 | 29 | 2018-12-01T00:00:00 | 2018-12-08T00:00:00 | 2711 | 34 | 1 | 167 | 115 | 28 |
| 9 | 3717 | 5 | 2018-12-01T00:00:00 | 2018-12-10T00:00:00 | 3763 | 69 | 2 | 753 | 376 | 32 |
| 10 | 4117 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 86 | 86 | 2 | 91 | 77 | 26 |

Source

Data store type External Workspace Sample dataset

Connection rosie.db.elephantsql.com;sybspbgi h... Refresh Edit + New

Connection type PostgreSQL Test connection Connection successful

Use query Table Query

Query * select * from sales_fact where salesdate > '@{variables('maxDate')}' Edit Preview data

Advanced

Additional columns New

Data

pipeline03_toWH01 | No label

Search

Trial:
59 days left

Home Activities Run View

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow

Lookups

Look_maxDate

Preview data

| | orderno | itemid | salesdate | deliverydate | customerid | cityid | qty | price | cost | discountpercent |
|----|---------|--------|---------------------|---------------------|------------|--------|-----|-------|------|-----------------|
| 1 | 1210 | 5 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 246 | 67 | 2 | 92 | 64 | 17 |
| 2 | 2130 | 8 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1457 | 96 | 1 | 52 | 36 | 11 |
| 3 | 2353 | 26 | 2018-12-01T00:00:00 | 2018-12-02T00:00:00 | 1327 | 40 | 1 | 167 | 115 | 4 |
| 4 | 2539 | 23 | 2018-12-01T00:00:00 | 2018-12-11T00:00:00 | 3569 | 59 | 1 | 83 | 66 | 2 |
| 5 | 3367 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 2470 | 51 | 1 | 67 | 53 | 34 |
| 6 | 3416 | 25 | 2018-12-01T00:00:00 | 2018-12-01T00:00:00 | 1907 | 84 | 2 | 753 | 376 | 25 |
| 7 | 3561 | 26 | 2018-12-01T00:00:00 | 2018-12-06T00:00:00 | 105 | 30 | 2 | 1333 | 666 | 34 |
| 8 | 3566 | 29 | 2018-12-01T00:00:00 | 2018-12-08T00:00:00 | 2711 | 34 | 1 | 167 | 115 | 28 |
| 9 | 3717 | 5 | 2018-12-01T00:00:00 | 2018-12-10T00:00:00 | 3763 | 69 | 2 | 753 | 376 | 32 |
| 10 | 4117 | 27 | 2018-12-01T00:00:00 | 2018-12-07T00:00:00 | 86 | 86 | 2 | 91 | 77 | 26 |

General Source Destination Mapping Settings

Data store type Workspace External

Workspace data store type

Data Warehouse Refresh Open

Table option Use existing Auto create table

Table Refresh Preview data Edit

> Advanced

Data Engineering

pipeline03_toWH01 | No label ▾

Search

Trial:
59 days left

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow

Lookup Look_maxDate

Set variable (x) set_maxDate

Copy data

Copy Data from SQL

Import schemas

+ New mapping

Preview source

Clear Reset Delete

Add dynamic content [Alt+Shift+D]

General Source Destination Mapping Settings

> Type conversion settings

Import schemas

+ New mapping

Preview source

Clear

Reset

Delete

Add dynamic content [Alt+Shift+D]

Data Engineering

pipeline03_toWH01 | No label

Search

Trial:
59 days left

Home Activities Run View

Validate Run Schedule

Lookup **Look_maxDate**

Set variable **(x) set_maxDate**

Please provide actual value of the parameters to get schema

Please provide actual value of the parameters for pipeline pipeline03_toWH01.

| Name | Type | Value |
|------------------|------|-------|
| No records found | | |

Additional expressions

| Name | Type | Value |
|-----------------------|--------|------------|
| @variables('maxDate') | string | 2018-01-01 |

General Source Destination Mapping Settings

> Type conversion settings

Import schemas

Add dynamic content [Alt+Shift+D]

Data Engineering

OK Cancel

- Input sample date value to preview data

pipeline03_toWH01 | No label

Search

Trial: 59 days left

Home Activities Run View

Lookup Look_maxDate → **Set variable** (x) set_maxDate → **Copy data** Copy Data from SQL

Mapping

| Source | Type | Destination | Type |
|--------------|----------|--------------|-----------|
| orderno | Int32 | orderno | int |
| itemid | Int32 | itemid | int |
| salesdate | DateTime | salesdate | datetime2 |
| deliverydate | DateTime | deliverydate | datetime2 |
| customerid | Int32 | customerid | int |
| cityid | Int32 | cityid | int |
| qty | Double | qty | float |

Data Engineering

Copying from column qty to column qty may have data truncation.

pipeline03_toWH01 | No label

Search

Trial: 59 days left

Home Activities Run View

Lookup Look_maxDate → **Set variable** (x) set_maxDate → **Copy data** Copy Data from SQL

Input

```
{
  "source": {
    "type": "PostgreSQLSource",
    "query": "select * from sales_fact where salesdate > 2018-12-01T00:00:00",
    "datasetSettings": {
      "type": "PostgreSQLTable",
      "schema": []
    }
  }
}
```

Pipe

| Activity | Run start | Duration | Activity run ID |
|--------------------|---------------------|--------------|--------------------------------------|
| Copy Data from SQL | 2023-08-26 14:34:02 | 39s | 02b13e02-f296-4a3e-8e27-18e6177656cc |
| set_maxDate | 2023-08-26 14:34:01 | Less than 1s | ca34a6db-49c4-4cee-b691-6a46882db66b |
| Look_maxDate | 2023-08-26 14:33:57 | 4s | ee33c579-0008-49f0-9cf5-a1cf10839323 |

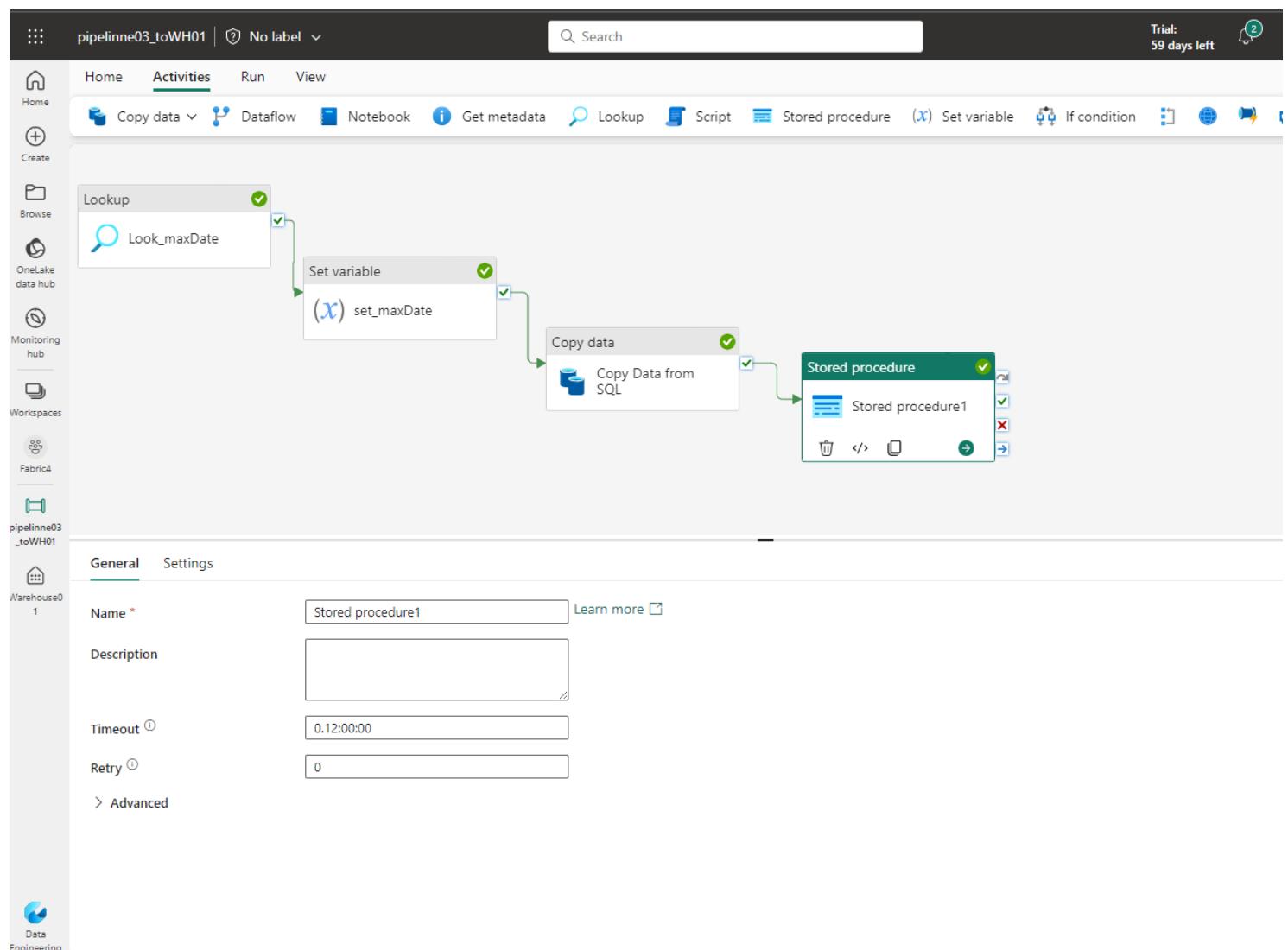
Pipeline status Succeeded

View run detail

Export to CSV

After complete the step, Save > validate > run to check if any error in the setup

Step 4: Stored procedure



pipeline03_toWH01 | No label

Search

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure Set variable If condition

Lookup Look_maxDate

Set variable (x) set_maxDate

Copy data Copy Data from SQL

Stored procedure Stored procedure1

```

graph LR
    L1[Lookup] --> S1[Set variable]
    S1 --> C1[Copy data]
    C1 --> SP1[Stored procedure]
  
```

General Settings

Data store type: Workspace

Data Warehouse: Warehouse01

Stored procedure name: [dbo].[Proc_GrossByDate]

Stored procedure parameters:

- Import
- New

pipeline03_toWH01 | No label

Search

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow Notebook Lookup Invoke pipeline

Lookup Look_maxDate

Set variable (x) set_maxDate

Copy data Copy Data from SQL

Stored procedure Stored procedure1

Parameters Variables Settings Output

Pipeline run ID: 5860e7c6-a738-4bb5-930a-be8cea648410

Pipeline status: Succeeded

| Activity name | Activity status | Activity type | Run start | Duration | Activity run ID |
|--------------------|-----------------|------------------|-----------------------|--------------|--------------------------------------|
| Stored procedure1 | Succeeded | Stored procedure | 8/26/2023, 3:00:06 PM | 5s | e9eee700-acd4-4fb8-8393-64b08aa59021 |
| Copy Data from SQL | Succeeded | Copy data | 8/26/2023, 2:59:24 PM | 41s | 00a6525a-549d-48a2-811b-257833a59fae |
| set_maxDate | Succeeded | Set variable | 8/26/2023, 2:59:23 PM | Less than 1s | 4acb86c1-0efd-4d48-8512-40c32089f269 |
| Look_maxDate | Succeeded | Lookup | 8/26/2023, 2:59:09 PM | 14s | 20040439-c799-4898-85f2-da5a129cd506 |

All status

Showing 1 - 4 items

Export to CSV

After complete the step, Save > validate > run to check if any error in the setup

2.7.3 Update data in SQL

The screenshot shows the ElephantSQL interface with the 'BROWSER' tab selected. The main area is titled 'SQL Browser' and contains the following SQL code:

```
-- SQL query 2
insert into sales_fact
select *
from sales
where salesdate > '2018-12-01' and salesdate <= '2018-12-31'
```

Below the code, there are two buttons: 'Table queries ▾' and 'Previous queries ▾'. To the right is a blue 'Execute ▶' button. A message at the bottom states 'No rows returned'. In the top right corner, a 'Fabric' dropdown menu is open, and a notifications panel shows a 'success' message: 'Query completed' with a green checkmark.

The screenshot shows the ElephantSQL interface with the 'BROWSER' tab selected. The main area is titled 'SQL Browser' and contains the following SQL code:

```
select count(*)
from sales_fact
```

Below the code, there are two buttons: 'Table queries ▾' and 'Previous queries ▾'. To the right is a blue 'Execute ▶' button. The results section shows the output of the query:

| count |
|-------|
| 4712 |

Run pipeline manually to load data

pipeline03_toWH01 | No label ▾

Search

Trial:
59 days left

Home Activities Run View

Validate Run Schedule View run history Copy data Dataflow Notebook Lookup

Lookup Look_maxDate

Set variable set_maxDate

Copy data Copy Data from SQL

Stored procedure Stored procedure1

Parameters Variables Settings Output

Pipeline run ID: 5860e7c6-a738-4bb5-930a-be8cea648410

Pipeline status Succeeded

All status Export to CSV

| Activity name | Activity status | Activity type | Run start | Duration | Activity run ID |
|--------------------|-----------------|------------------|-----------------------|--------------|--------------------------------------|
| Stored procedure1 | Succeeded | Stored procedure | 8/26/2023, 3:00:06 PM | 5s | e9eee700-acd4-4f8b-8393-64b08aa59021 |
| Copy Data from SQL | Succeeded | Copy data | 8/26/2023, 2:59:24 PM | 41s | 00a6525a-549d-48a2-811b-257833a59fae |
| set_maxDate | Succeeded | Set variable | 8/26/2023, 2:59:23 PM | Less than 1s | 4acb86c1-0efd-4d48-8512-40c32089f269 |
| Look_maxDate | Succeeded | Lookup | 8/26/2023, 2:59:09 PM | 14s | 20040439-c799-4898-85f2-da5a129cd506 |

Pipeline: pipeline03_toWH01 | No label

Search:

Trial: 59 days left

Home Activities Run View

Validate Cancel Schedule View run history Copy data Dataflow Notebook

Lookup: Look_maxDate → **Set variable**: (x) set_maxDate → **Copy data**: Copy Data from SQL → **Stored procedure**: Stored procedure1

Output

Pipeline run ID: 09639d89-9acb-4dc9-b758-24d12862aca7 Pipeline status: In progress

| Activity name | Activity status | Activity type | Run start | Duration | Activity run ID |
|--------------------|-----------------|------------------|-----------------------|--------------|--------------------------------------|
| Stored procedure1 | Succeeded | Stored procedure | 8/26/2023, 3:09:39 PM | 4s | 9d20582d-ee8c-4763-9398-dcfad67a8ccb |
| Copy Data from SQL | Succeeded | Copy data | 8/26/2023, 3:08:56 PM | 43s | c2c74699-b7c0-4b85-b1b7-bebd82e88824 |
| set_maxDate | Succeeded | Set variable | 8/26/2023, 3:08:55 PM | Less than 1s | 4c702704-4c56-4c98-a268-8a87a6b1193c |
| Look_maxDate | Succeeded | Lookup | 8/26/2023, 3:08:51 PM | 4s | 230cd85e-ae88-4ab0-a6b0-27cec00e2744 |

Check data in WH

Warehouse01 | No label

Search:

Trial: 59 days left

Home Get data New SQL query New visual query

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Explorer

- + Warehouses
 - dbo
 - Tables
 - GrossByDate
 - sales
 - Views
 - Functions
 - Stored Procedures
 - Proc_GrossByDate
 - guest
 - INFORMATION_SCHEMA

SQL query 6

```
1 select count(*)
2 from sales
3
4 SELECT max(salesdate) maxDate
5 from GrossByDate
```

Results

| | 123 | untitled1 |
|---|------|-----------|
| 1 | 4712 | |

The screenshot shows the Snowflake UI interface. On the left, there's a sidebar with icons for Home, Create, Browse, Workspaces, Fabric4, and Warehouse 01. The main area has a "Home" tab selected. At the top, it says "Warehouse01 | No label". There's a search bar and a trial status "Trial: 59 days left". Below the search bar are buttons for "Get data", "New SQL query", and "New visual query". A message states: "A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)". The "Explorer" pane on the left shows a tree view of the database schema under "Warehouses". The "dbo" node contains "Tables" (GrossByDate, sales), "Views", "Functions", "Stored Procedures" (Proc_GrossByDate), "guest", "INFORMATION_SCHEMA", "sys", and "Security". The "Messages" tab shows a query history:

```
1 select count(*)  
2 from sales  
3  
4 SELECT max(salesdate) maxDate  
5 from GrossByDate
```

The "Results" tab shows the output of the last query:

| | maxDate |
|---|----------------------------|
| 1 | 2018-12-31 00:00:00.000000 |

Load new data to sales_fact table

The screenshot shows the ElephantSQL UI. On the left, there's a sidebar with links: DETAILS, ALARMS, BROWSER (which is selected), STATS, SLOW QUERIES, BACKUP, LOG, METRICS, ADMIN, INTEGRATIONS, FIREWALL, and MAINTENANCE. The main area is titled "SQL Browser". It shows a query editor with the following SQL code:

```
-- SQL query 3  
insert into sales_fact  
select *  
from sales  
where salesdate > '2018-12-31' and salesdate <= '2019-02-01'
```

Below the editor are buttons for "Table queries" and "Previous queries". To the right, there's a "Notifications" panel with a "success" message: "Query completed" with a green checkmark. At the bottom right is a blue "Execute" button.

DETAILS

ALARMS

BROWSER

STATS

SLOW QUERIES

BACKUP

LOG

METRICS

ADMIN

INTEGRATIONS

FIREWALL

MAINTENANCE

SQL Browser

```
select count(*)
from sales_fact
```

Table queries ▾ Previous queries ▾

count
6623

Execute ►

Notifications Dismiss all

Success Query completed

Run manual pipeline

pipeline03_toWH01 | ? No label ▾

Search

Trial: 59 days left

Home Activities Run View

Home Create Browse OneLake data hub Monitoring hub Workspaces Fabric4 pipeline03_toWH01 Warehouse0

Lookup Look_maxDate → Set variable (x) set_maxDate → Copy data Copy Data from SQL → Stored procedure Stored procedure1

Validate Cancel Schedule View run history Copy data Dataflow Notebook

Parameters Variables Settings Output

Pipeline run ID: b922252e-09eb-403d-82fe-d4cbc0282516 ⏪ Pipeline status In progress

All status ▾

Showing 1 - 4 items

| Activity name ↑↓ | Activity status ↑↓ | Activity type ↑↓ | Run start ↑↓ | Duration ↑↓ | Activity run ID |
|--------------------|--------------------|------------------|-----------------------|--------------|-----------------|
| Stored procedure1 | ✓ Succeeded | Stored procedure | 8/26/2023, 3:16:14 PM | 5s | 7d068c50-f2a0-4 |
| Copy Data from SQL | ✓ Succeeded | Copy data | 8/26/2023, 3:15:33 PM | 40s | fcd0ae0a-7466-4 |
| set_maxDate | ✓ Succeeded | Set variable | 8/26/2023, 3:15:32 PM | Less than 1s | b2d2743c-5bde- |
| Look_maxDate | ✓ Succeeded | Lookup | 8/26/2023, 3:15:27 PM | 4s | 85d59abb-67d3- |

Warehouse01 | ? No label ▾

Search

Trial:
59 days left

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

Warehouse 01

Home

Get data ▾

New SQL query ▾

New visual query

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Explorer

+ Warehouses

- dbo
 - Tables
 - GrossByDate
 - sales
 - Views
 - Functions
 - Stored Procedures
 - Proc_GrossByDate
 - guest
 - INFORMATION_SCHEMA
 - sys
- Security

SQL query 6

Run Save as view

```
1 select count(*)  
2 from sales  
3  
4 SELECT max(salesdate) maxDate  
5 from GrossByDate
```

Messages Results Save as table ...

| 123 | untitled1 |
|-----|-----------|
| 1 | 6623 |

Warehouse01 | ? No label ▾

Search

Home

Create

Browse

OneLake data hub

Monitoring hub

Workspaces

Fabric4

Warehouse 01

Home

Get data ▾

New SQL query ▾

New visual query

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Explorer

+ Warehouses

- dbo
 - Tables
 - GrossByDate
 - sales
 - Views
 - Functions
 - Stored Procedures
 - Proc_GrossByDate
 - guest
 - INFORMATION_SCHEMA
 - sys
- Security

SQL query 6

Run Save as view

```
1 select count(*)  
2 from sales  
3  
4 SELECT max(salesdate) maxDate  
5 from GrossByDate
```

Messages Results Save as table ...

| 1 | maxDate |
|---|----------------------------|
| 1 | 2019-02-01 00:00:00.000000 |

Make data double in SQL database

Create new data

The screenshot shows the ElephantSQL interface with the "BROWSER" tab selected. A SQL query is entered in the main area:

```
-- SQL query 4
insert into sales_fact
select *
from sales
where salesdate > '2019-02-01' and salesdate <= '2019-04-01'
```

The results section below the query editor displays the message "No rows returned".

The screenshot shows the ElephantSQL interface with the "BROWSER" tab selected. A SQL query is entered in the main area:

```
select count(*)
from sales_fact
```

A notification window titled "Notifications" appears in the top right corner, indicating "Query completed" with a green checkmark.

The results section below the query editor displays the output:

| count |
|-------|
| 10202 |

Input query to make duplicated data

The screenshot shows the ElephantSQL interface with the 'BROWSER' tab selected. In the main query editor, the following SQL code is entered:

```
insert into sales_fact
select *
from sales
where salesdate > '2018-12-31' and salesdate <= '2019-02-01'
```

The 'Notifications' panel indicates a 'success' message: 'Query completed'. The results section shows 'No rows returned'.

The screenshot shows the ElephantSQL interface with the 'BROWSER' tab selected. In the main query editor, the following SQL code is entered:

```
select count(*)
from sales_fact
```

The 'Notifications' panel indicates a 'success' message: 'Query completed'. The results section displays the output:

| count |
|-------|
| 12113 |

pipeline03_toWH01 | No label

Search

Trial: 59 days left

Run

Lookup
Look_maxDate

Set variable
(x) set_maxDate

Copy data
Copy Data from SQL

Stored procedure
Stored procedure1

Parameters Variables Settings Output

Pipeline run ID: 5775b8f5-6fab-4ce6-a5ee-97be1f725749

Pipeline status: Succeeded

All status

Showing 1 - 4 items

| Activity name | Activity status | Activity type | Run start | Duration | Activity run ID |
|--------------------|-----------------|------------------|-----------------------|--------------|--------------------------------------|
| Stored procedure1 | Succeeded | Stored procedure | 8/26/2023, 3:21:24 PM | 6s | 807f51f6-5b00-406e-83c4-1ba6961ec538 |
| Copy Data from SQL | Succeeded | Copy data | 8/26/2023, 3:20:42 PM | 41s | 3cca576e-59b2-4fc9-8a01-eec95e52e61f |
| set_maxDate | Succeeded | Set variable | 8/26/2023, 3:20:42 PM | Less than 1s | 2a7f4740-2ccd-475e-b466-c084a684dfb0 |
| Look_maxDate | Succeeded | Lookup | 8/26/2023, 3:20:37 PM | 4s | 70df4bf5-20b8-4294-867d-335e65961bb0 |

Duplicated data is not brought to Fabric

Warehouse01 | No label

Search

Home

Get data | New SQL query | New visual query

A default dataset for faster reporting was created and will be automatically updated with any tables and views added to the warehouse. [Learn more](#)

Explorer

- + Warehouses
 - dbo
 - Tables
 - GrossByDate
 - sales
 - Views
 - Functions
 - Stored Procedures
 - Proc_GrossByDate
 - guest
 - INFORMATION_SCHEMA
 - sys
 - Security

SQL Browser

```
select count(*)
from sales_fact
```

Table queries | Previous queries

count

12113

Setup batch job recurrence for running Pipeline automatically

The screenshot shows the Microsoft Fabric4 workspace interface. On the left, there's a vertical sidebar with icons for Home, Create, Browse, OneLake data hub, Monitoring hub, Workspaces (selected), and Data Warehouse. The main area displays a list of datasets and pipelines under the 'Fabric4' fabric. A context menu is open over the 'pipeline01_to wh01' entry, which is highlighted with a yellow background. The menu items are:

- Open
- Delete
- Settings
- Add to Favorites
- View lineage
- View details
- Schedule (highlighted)
- Recent runs
- Save as

| Name | Type | Owner |
|-------------------------------|-------------------|-----------|
| LakeHouse10 | Lakehouse | Hien Tran |
| LakeHouse10 (1) | | |
| Notebook 1 | | |
| Notebook 2 | | |
| Notebook 3 | | |
| Notebook-3 | | |
| NoteBook-Fabric4- LakeHouse10 | | |
| pipeline01- to wh01 | | |
| pipeline03_toWH01 | Data pipeline | Hien Tran |
| PowerBI4 | Report | Fabric4 |
| Report_Iris | Report | Fabric4 |
| Sales | Dashboard | Fabric4 |
| Warehouse01 | Dataset (default) | Fabric4 |
| Warehouse01 | Warehouse | Hien Tran |



Home



Fabric4

+ New

Upload



OneLake

data hub



Fabric4



Data

Warehouse



Fabric4

Name

LakeHouse10

LakeHouse10 (1)

Notebook 1

Notebook 2

Notebook 3

Notebook-3

NoteBook-Fabric4- LakeHouse10

pipeline01- to wh01

pipeline03_toWH01

PowerBI4

Report_Iris

Sales

Warehouse01

Warehouse01

pipeline03_toWH01
Data pipeline

About

Last success is in
August 26, 2023 at 8:20:34 AM
(UTC) Coordinated Universal Time

Sensitivity label

The scheduled refresh is turned off

Endorsement

Run

Schedule

Schedule

Scheduled run

 On Off

Repeat

Daily

Time

01:00 AM

Add a time

Start

08/27/2023 12:00 AM

End

08/31/2023 12:25 PM

Time zone

(UTC+07:00) Bangkok, Hanoi, Jakarta

Apply

Discard