

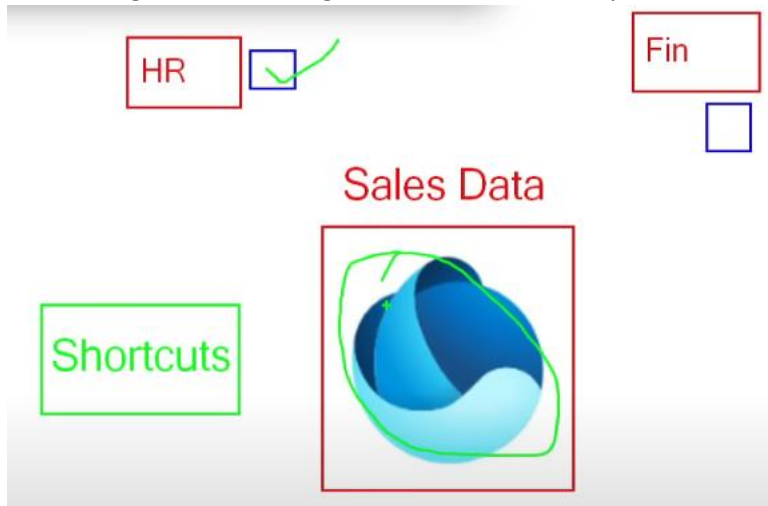
FABRIC DATA FACTORY

MF is End to End Unified Data Solution (from data ingestion to deployment)

Best part of MS is that all the services are connected/integrated (we don't have to worry about connections)

Fabric Data Factory → ETL/ELT solution of Fabric, helps in Data Movement, Transformation, Orchestration and Integration

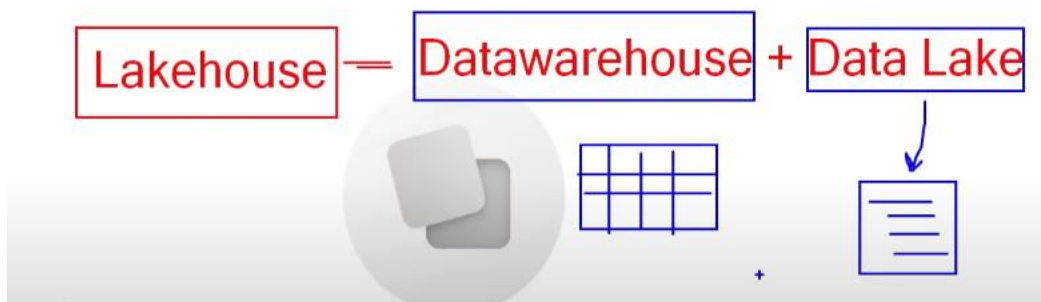
OneLake is cheaper than ADLS Gen2 since we have only one copy of data, no need to manage networking between storage accounts since its only one lake.



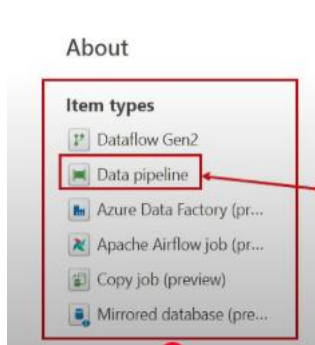
Let's say we have Sales data, now we have 2 workspaces HR and Finance, how the Sales data made available to both the workspaces? OL bts creates Shortcuts

to see content of OL

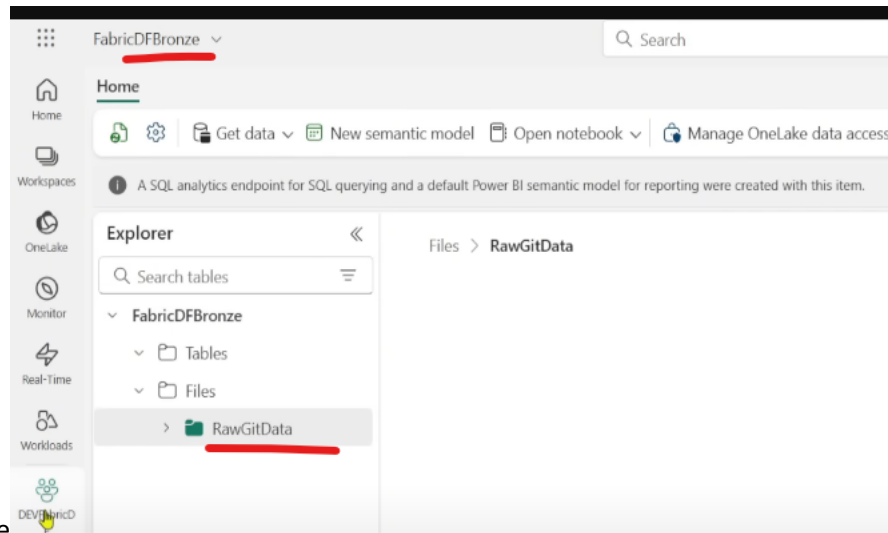
Name	Type	Owner	Refreshed	Location	Endorsement
anshlakehousesilv...	Lakehouse	fabricut	—	anshLambaprows	—
anshlakeansh	Lakehouse	fabricut	—	anshLambaprows	—
anshlakeansh	SQL analytics endpoint	fabricut	—	anshLambaprows	—
anshsemantic	Semantic model	fabricde	19/1/25, 7:44:27 pm	CICDWS	—
SilverLH	Semantic model (default)	fabricde	19/1/25, 7:44:23 pm	CICDWS	—
SilverLH	SQL analytics endpoint	fabricde	—	CICDWS	—



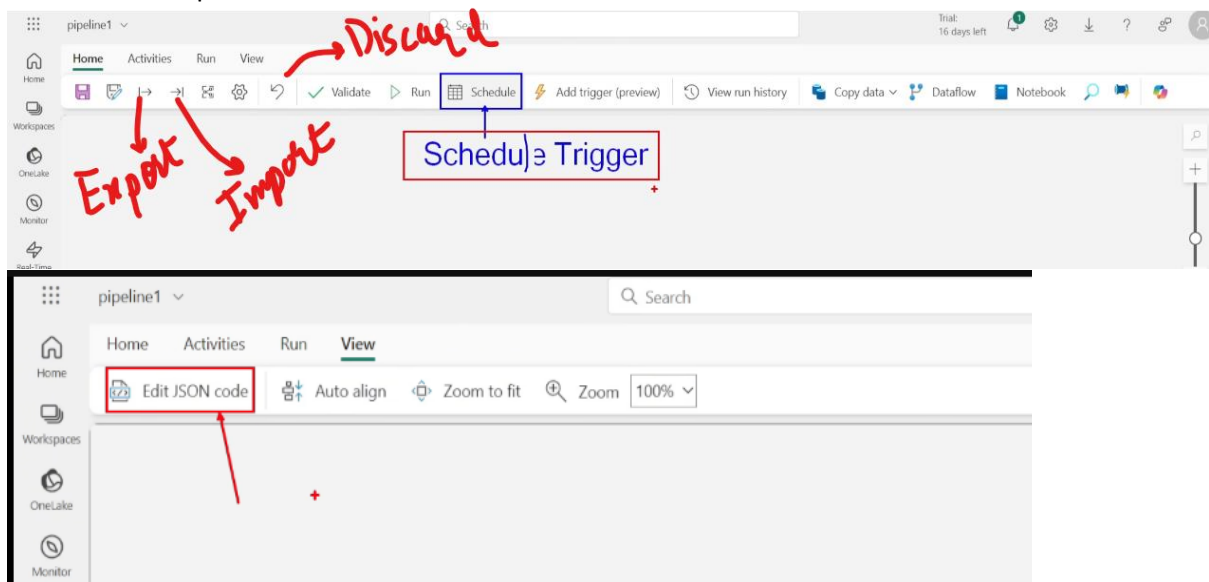
Simply put files in Data Lake, DWH is a logical DWH over here, it will create a metadata layer on top of files and it will behave like a table



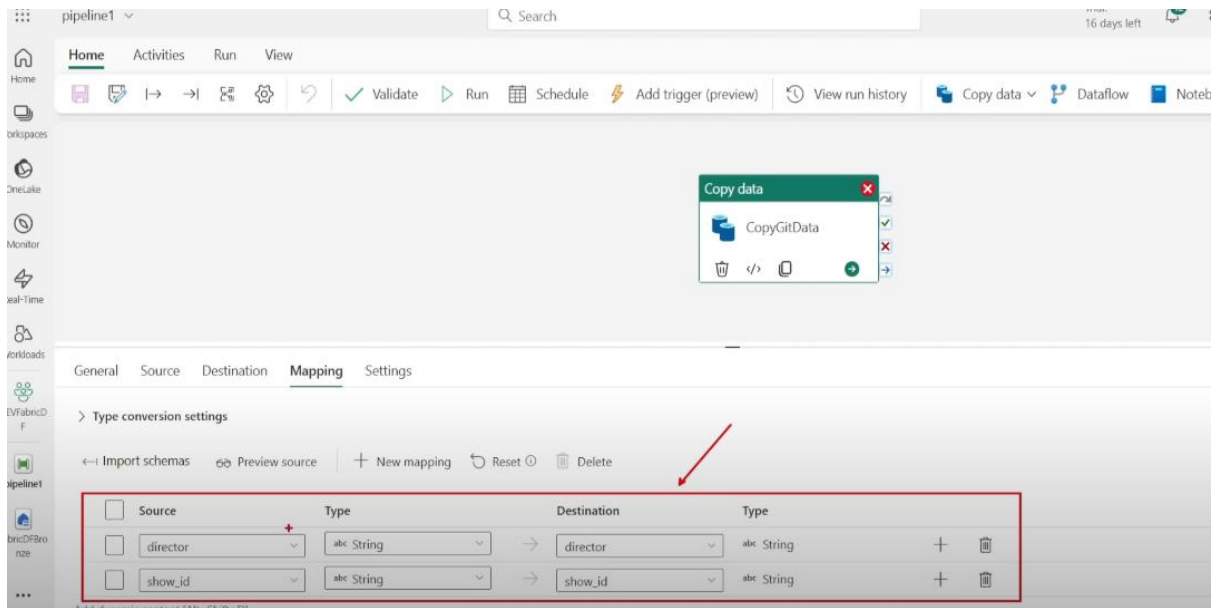
Go to app.fabric.microsoft.com
Create a Workspace



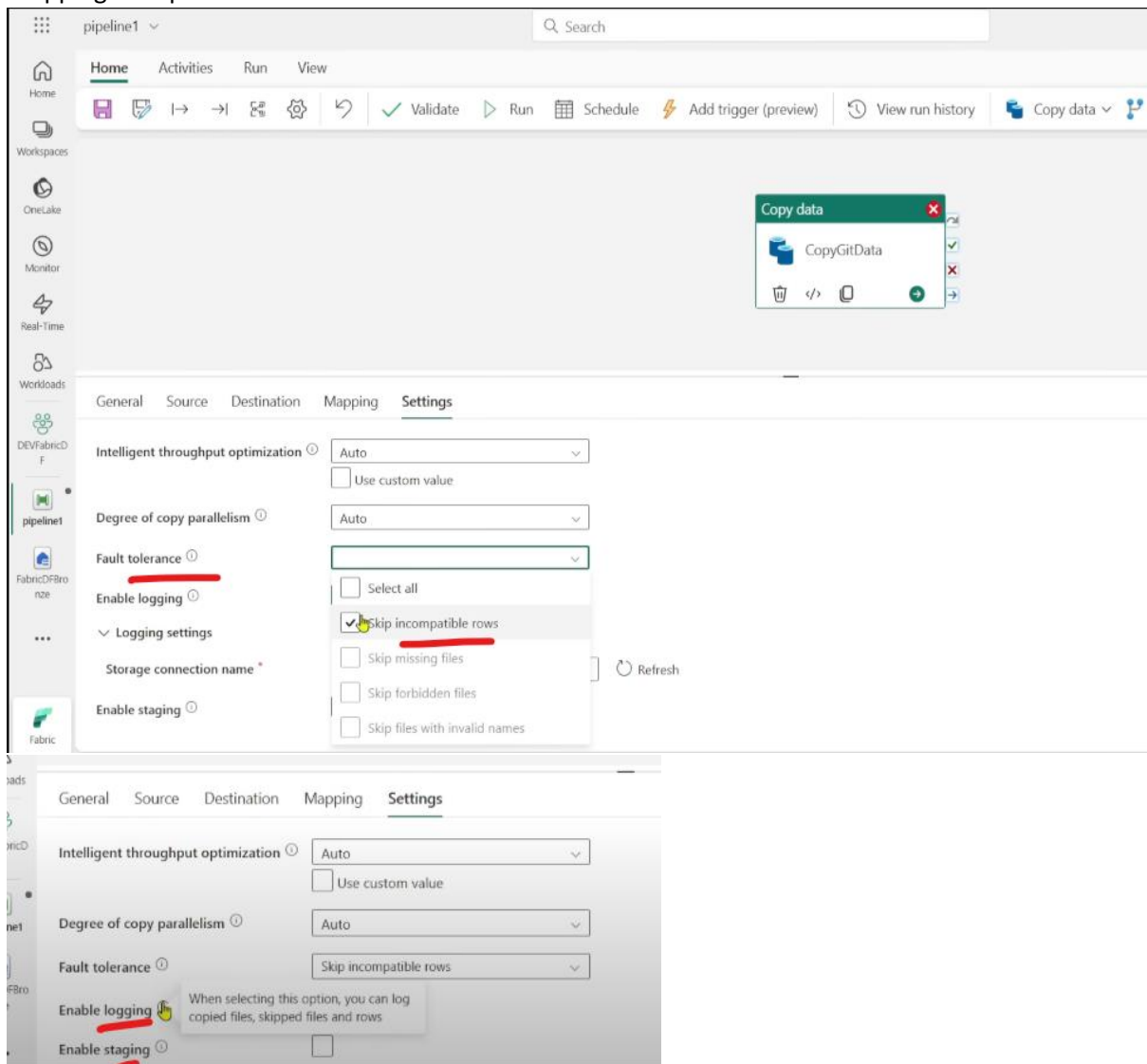
Create a Lakehouse
Create a Data Pipeline

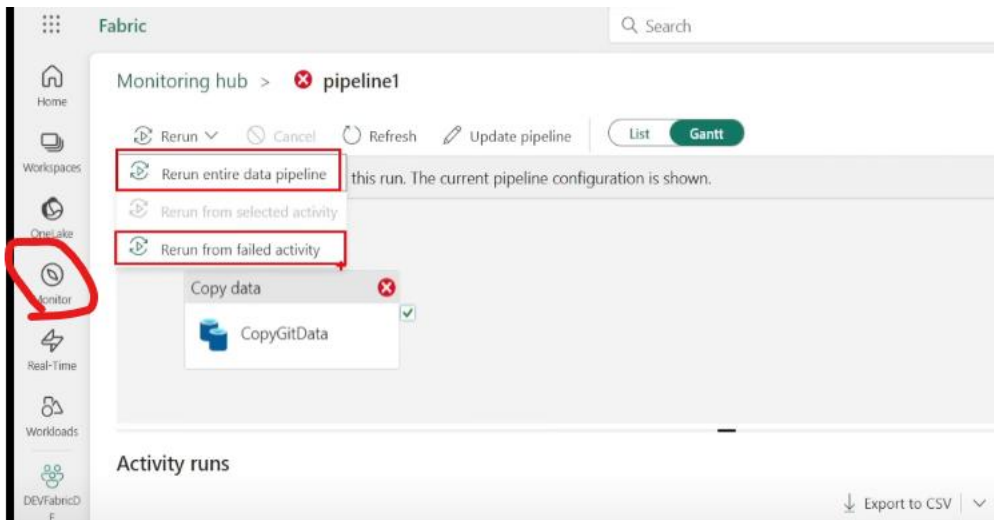


Get data from GitHub and store in LH

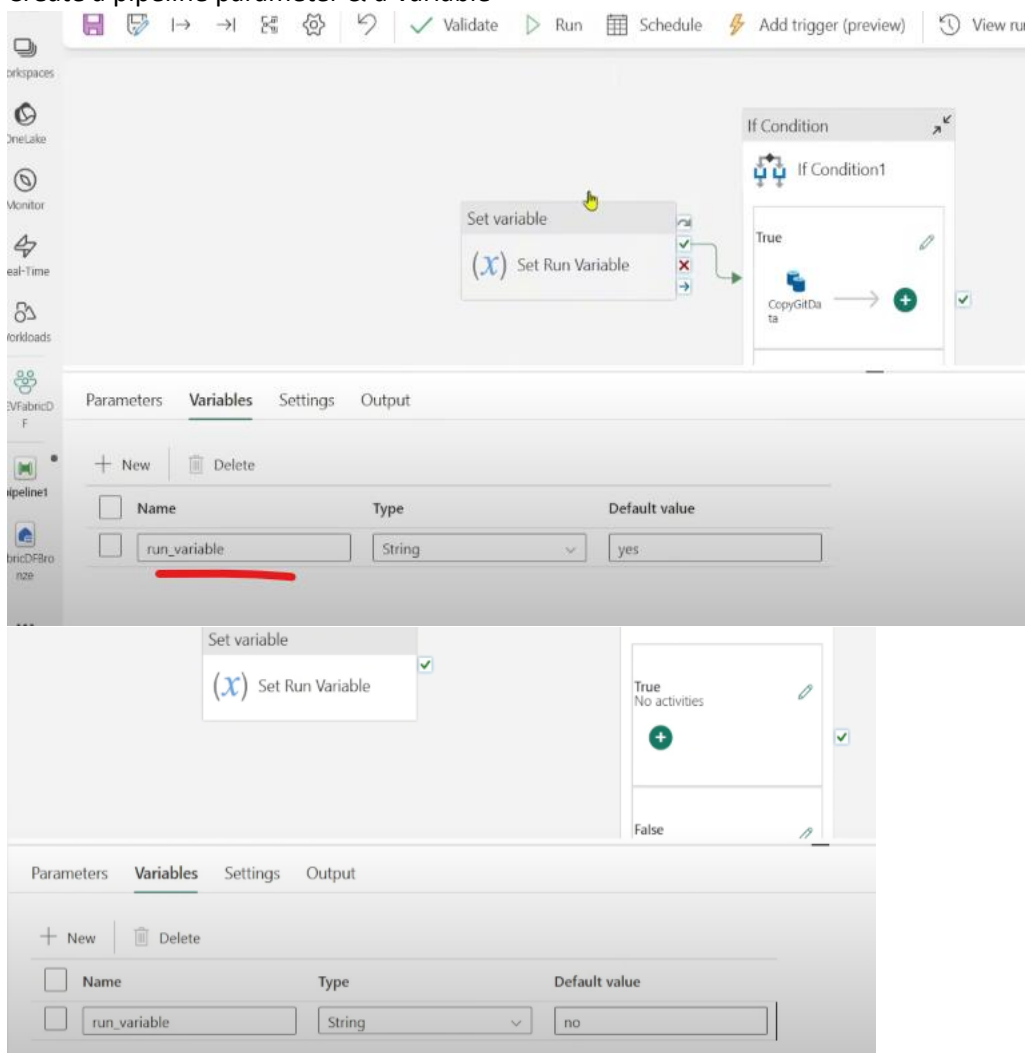


Mapping is required for Schema Enforcement





Problem statement1: Copy data from GitHub only if the pipeline parameter is yes
Create a pipeline parameter & a Variable



pipeline1

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure (x)

Set variable

(x) Set Run Variable

If Condition

If Condition1

True No activities

False

Copy data

CopyGitData

General Settings

Variable type Pipeline variable Pipeline return value

Name run_variable + New

Value @pipeline().parameters.run_flag

pipeline1

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure (x)

Set variable

(x) Set Run Variable

If Condition

If Condition1

True No activities

False No activities

General Activities (0)

Expression @equals(variables('run_variable'),'yes')

Case	Activity
True	No activities
False	No activities

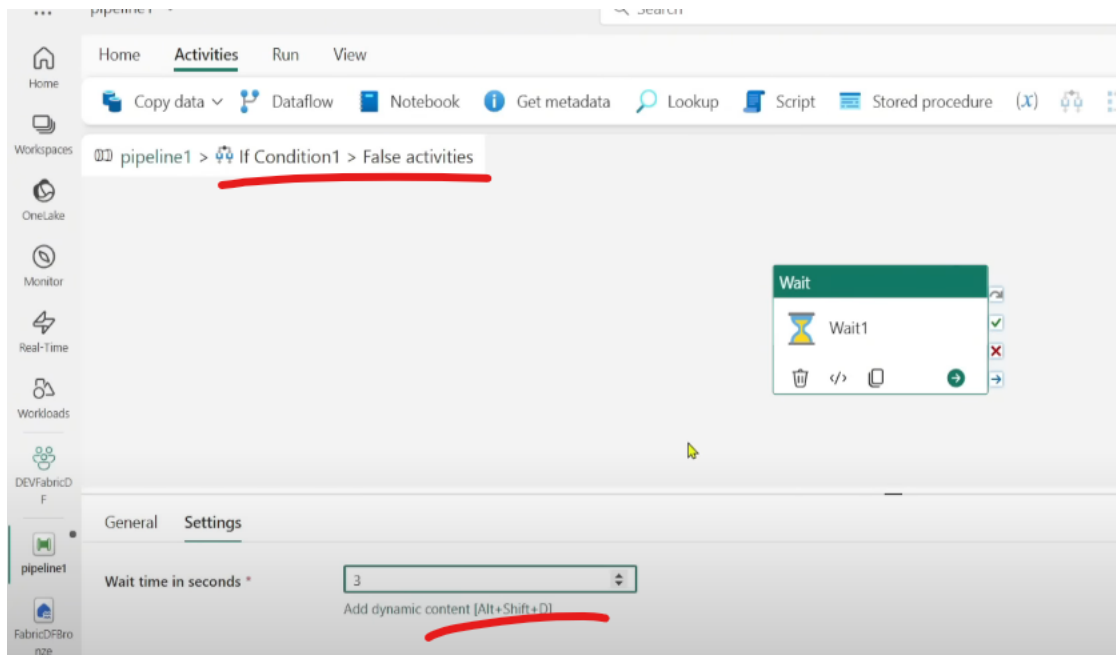
Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure (x)

pipeline1 > If Condition1 > True activities

Copy data

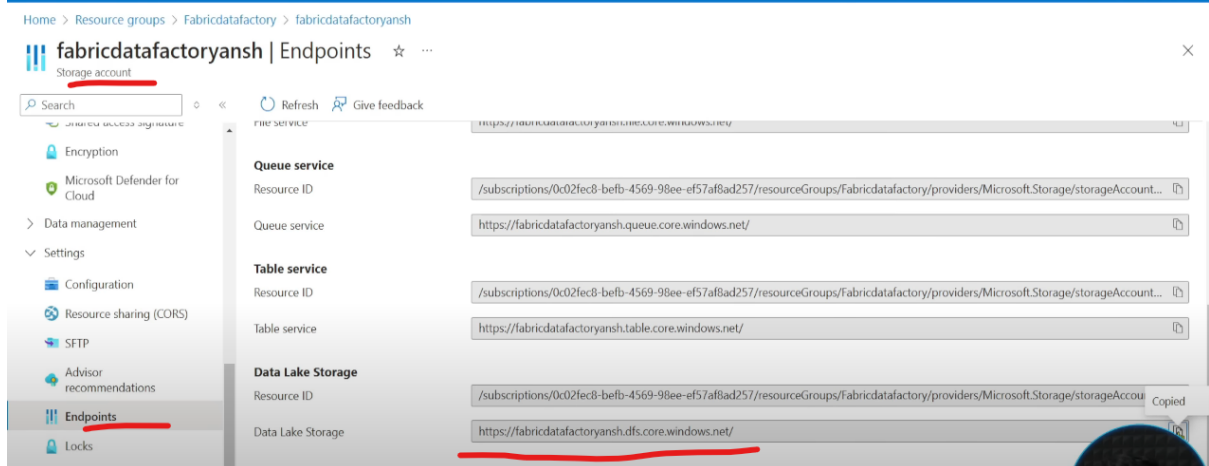
CopyGitData



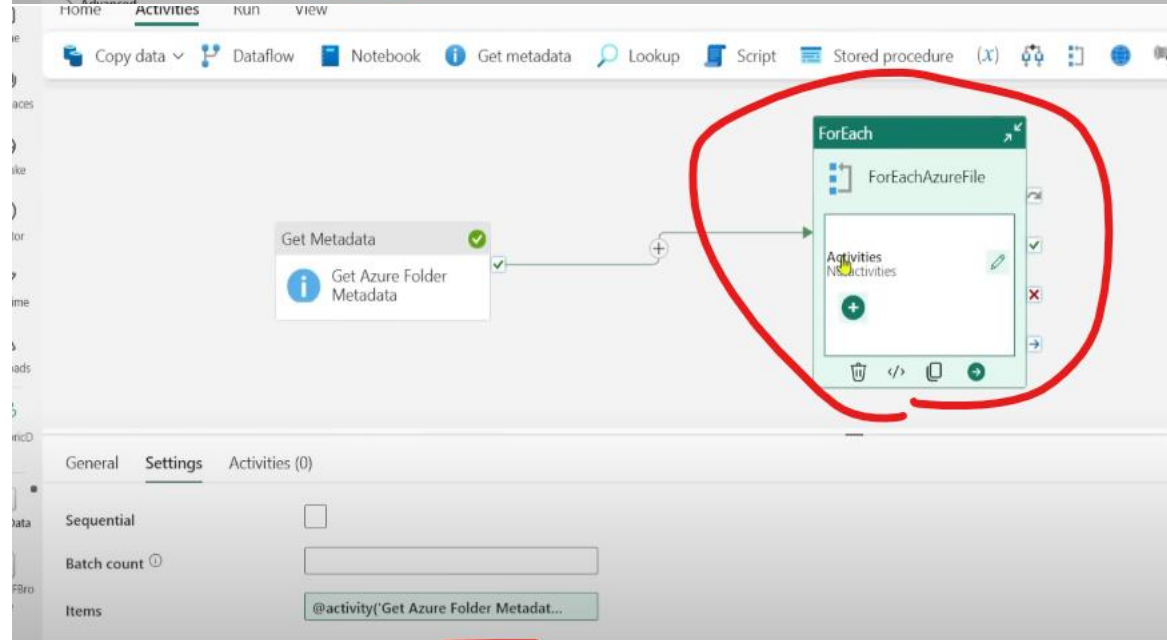
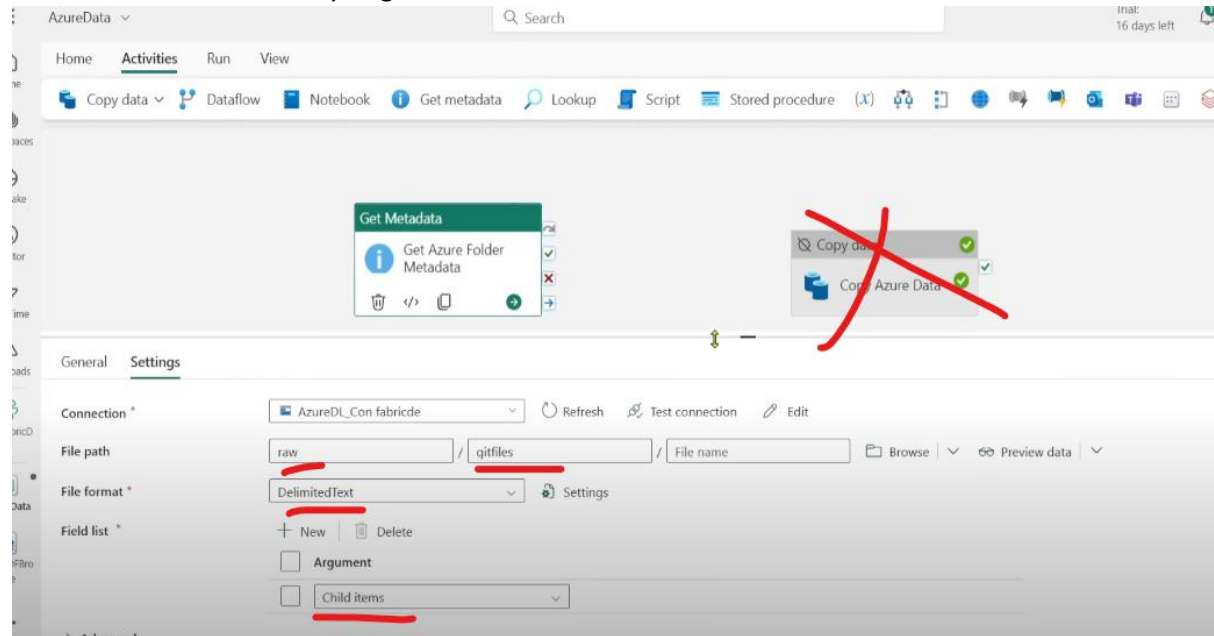
Note : We cant use For-Each inside If condition in Fabric (in ADF its supported)

Problem statement2: Get a list of files from ADLS and keep it in Fabric LH

Give Storage Blob Data Contributor access of ADLS to Fabric



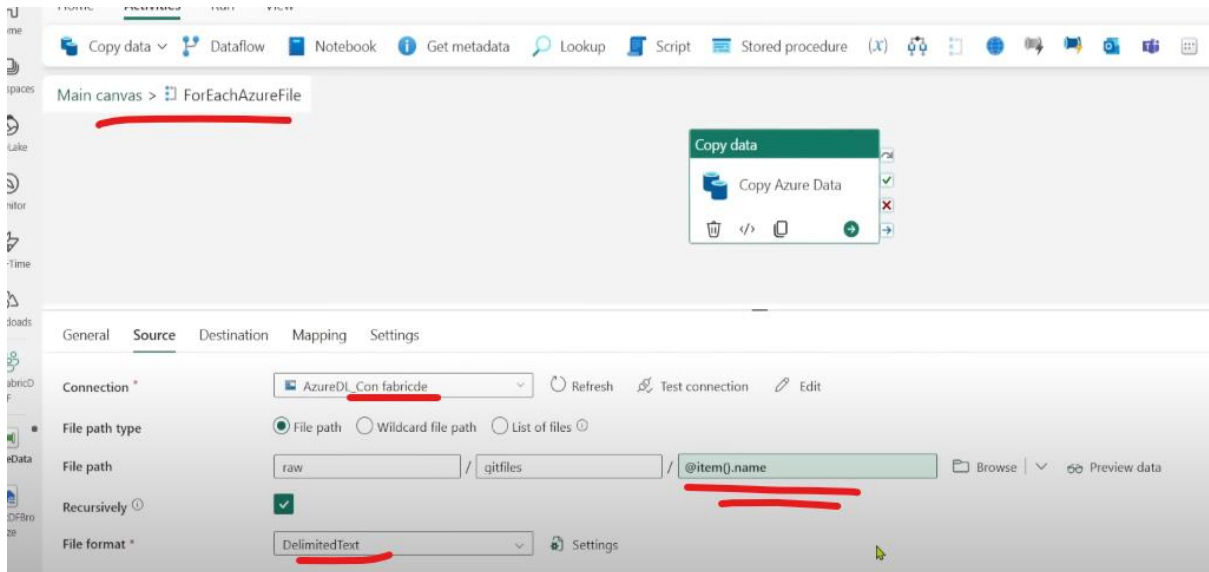
Use a Get Metadata activity to get all the File names



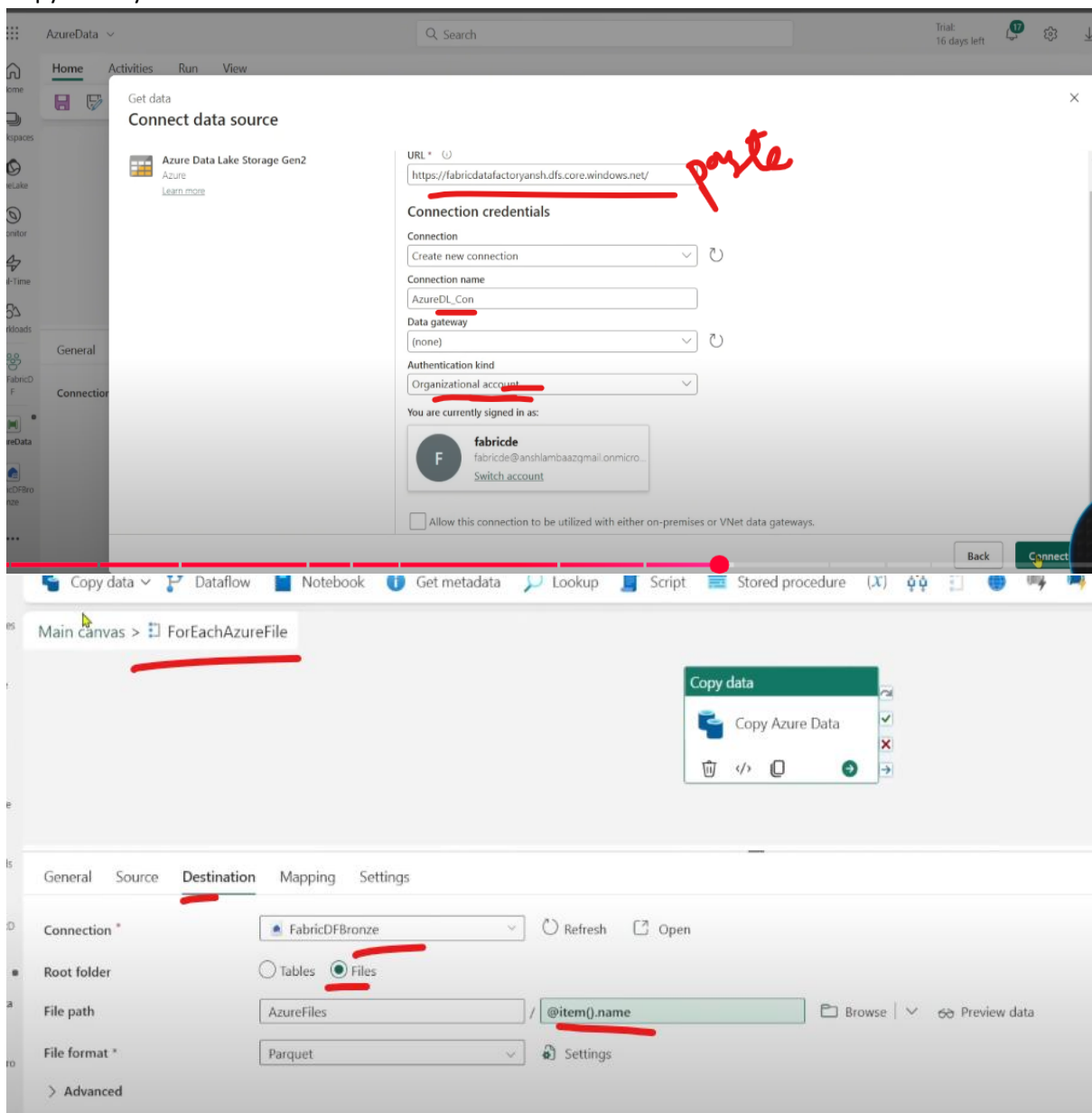
Pipeline expression builder

Add dynamic content below using any combination of expressions, functions and sy

@activity('Get Azure Folder Metadata').output.childItems

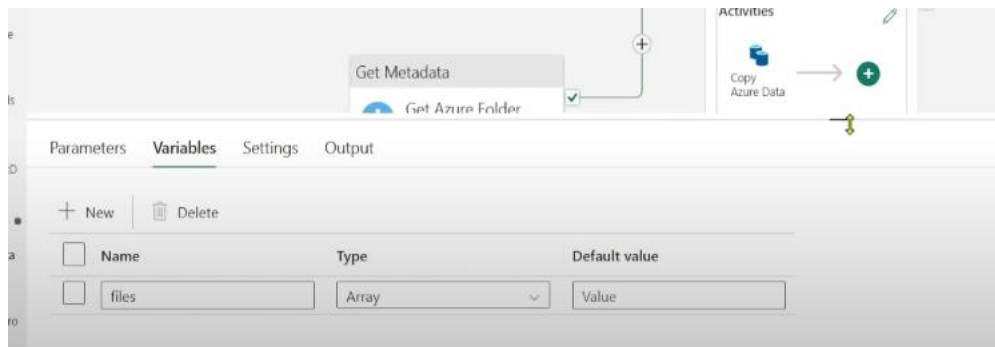


Copy activity → Source → connection for ADLS Gen2

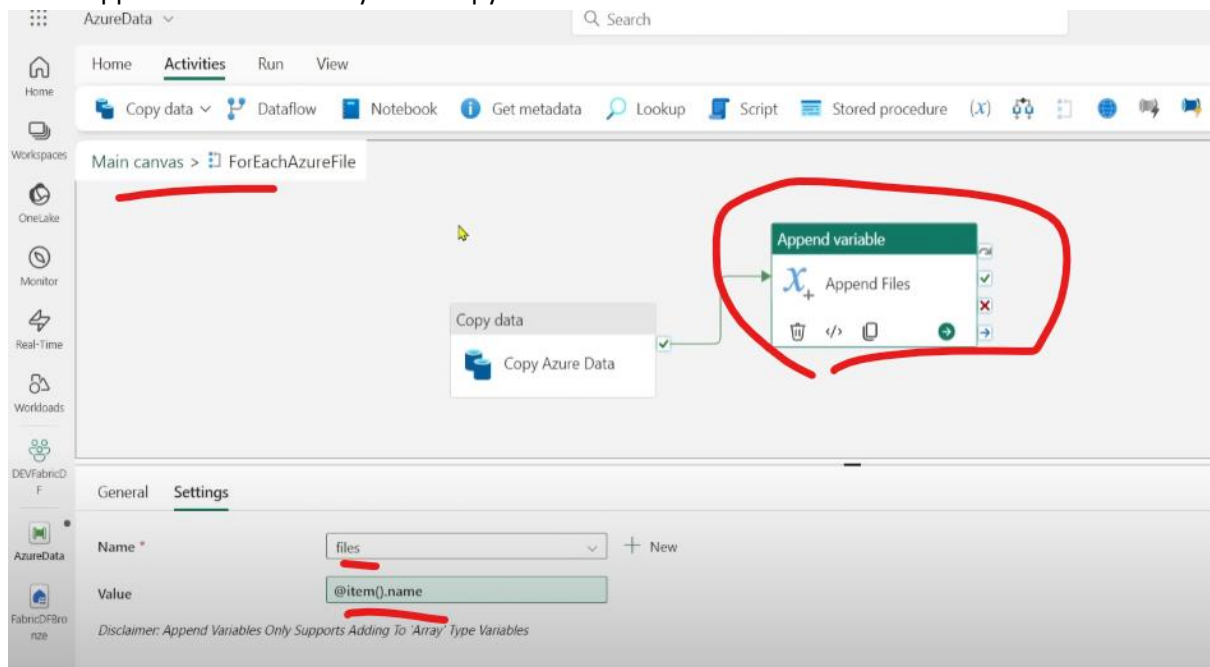


Lets enhance the pipeline by adding logging mechanism

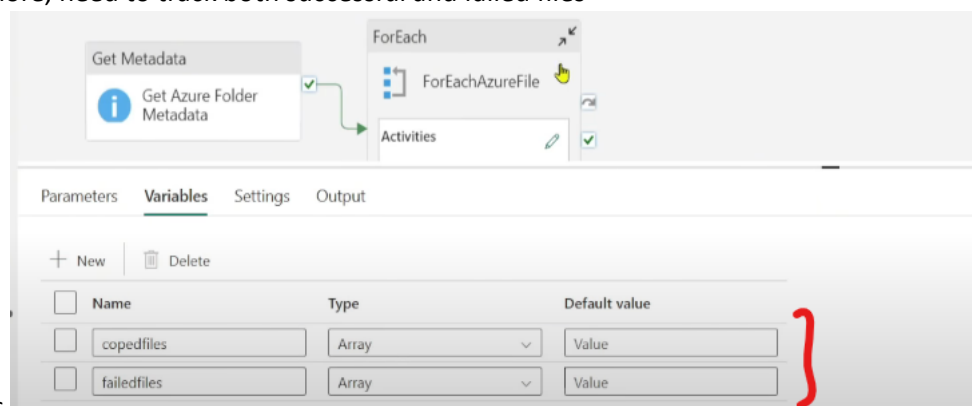
Create a variable, everytime the files are getting copied, the information needs to be stored in this variable



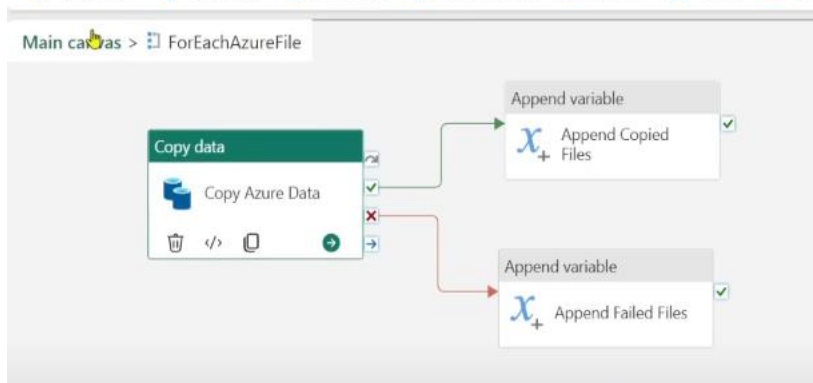
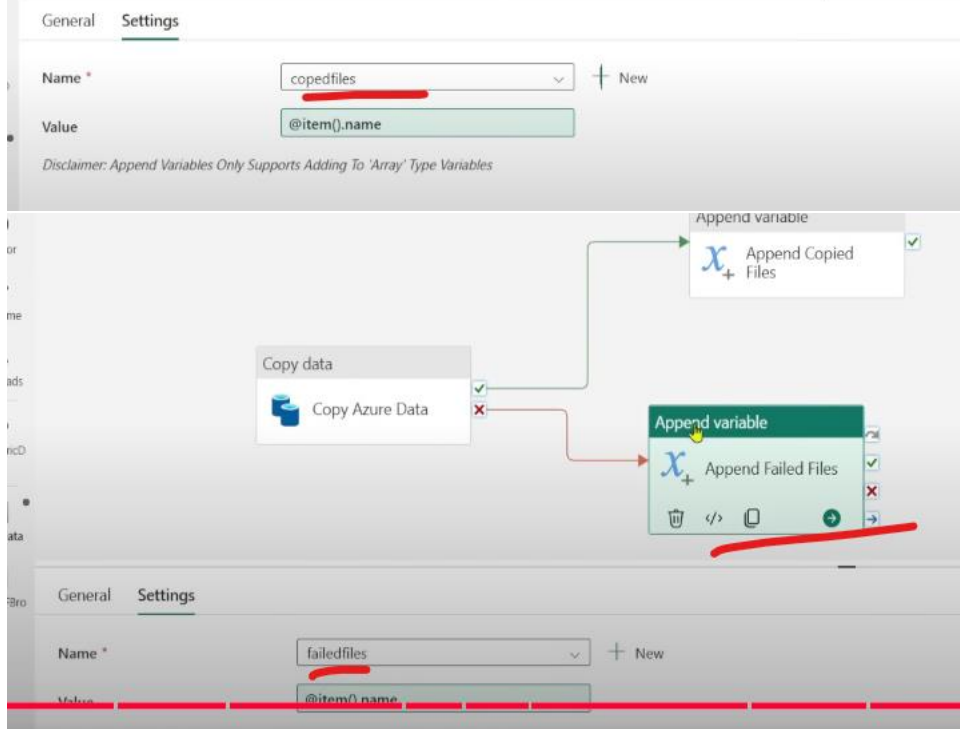
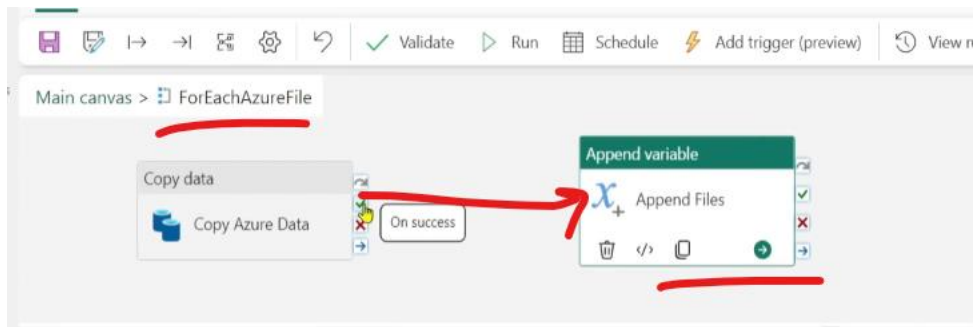
Add a Append variable activity after Copy to hold all the names



Lets enhance it more, need to track both successful and failed files



create 2 variables



the data of 2 arrays inside the loop also needs to be stored, create 2 variables for that

Get Metadata

Get Azure Folder

ForEach

ForEachAzureFile

Activities

Copy Azure Data

Append Copied

Parameters Variables Settings Output

Name	Type	Default value
copedfiles	Array	Value
failedfiles	Array	Value
copedfilesSIZE	Integer	Value
failedfilesSIZE	Integer	Value

AzureData

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure

Workspaces

OneLake

Monitor

Real-Time

Workloads

DEVFabricD F

AzureData

FabricDFBro nze

Get Metadata

Get Azure Folder Metadata

ForEach

ForEachAzureFile

Activities

Copy Azure Data

Append Copied

Set variable

(X) CopedFilesSize

Variable type

Pipeline variable Pipeline return value

Name *

copedfilesSIZE

Value

@length(variables('copedfiles'))

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure

Workspaces

OneLake

Monitor

Real-Time

Workloads

DEVFabricD F

AzureData

FabricDFBro nze

Get Metadata

Get Azure Folder Metadata

ForEach

ForEachAzureFile

Activities

Copy Azure Data

Append Copied

Set variable

(X) CopedFilesSize

Set variable

(X) FailedFilesSize

Variable type

Pipeline variable Pipeline return value

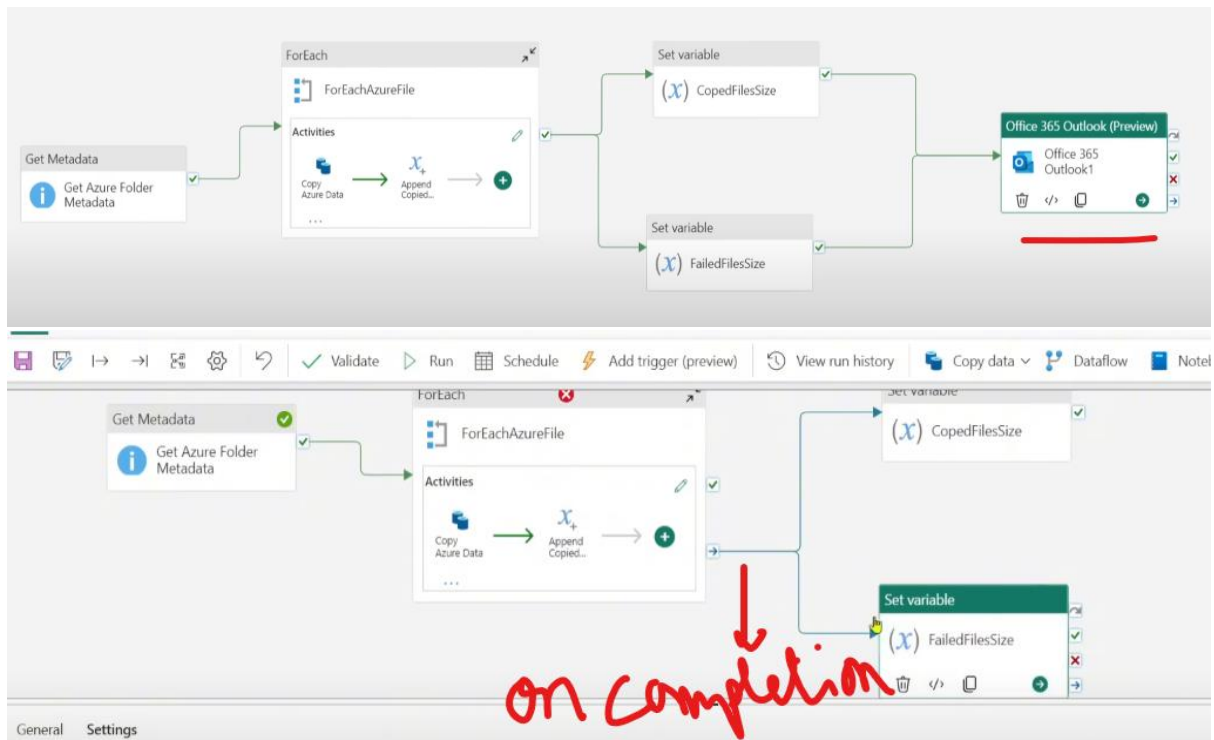
Name *

failedfilesSIZE

Value

@length(variables('failedfiles'))

Now add Notifications for these activities



Just a change, link both the Set Variable activities with On completion with For_each (not with success)

Delete the last activity and run the pipeline

Activity name	Activity status	Run start
Get Azure Folder Metadata	Succeeded	3/2/2025
ForEachAzureFile	Failed	3/2/2025
Copy Azure Data	Failed	3/2/2025
Copy Azure Data	Failed	3/2/2025

Because of bad data

Output

Copy to clipboard

```
{
  "name": "copiedfilesSIZE",
  "value": 2
}
```

Output

Copy to clipboard

```
{
  "name": "failedfilesSIZE",
  "value": 2
}
```

I also want to store the details of failed names, create 2 more variables

Name	Type	Default value
copedfiles	Array	Value
failedfiles	Array	Value
copiedfilesSIZE	Integer	Value
failedfilesSIZE	Integer	Value
copiednames	String	Value
failednames	String	Value

AzureData

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure (x)

Set variable (x) CopiedNames

Set variable (x) CopiedFilesSize

ForEach

ForEachAzureFile

Activities

Copy Azure Data

Append Copied...

Set variable

General Settings

Variable type Pipeline variable Pipeline return value

Name copiednames

Value @variables('copiedfiles')

Home Activities Run View

Copy data Dataflow Notebook Get metadata Lookup Script Stored procedure (x)

Set variable (x) FailedFilesSize

Set variable (x) FailedNames

General Settings

Variable type Pipeline variable Pipeline return value

Name failednames

Value @variables('failedfiles')

Get Metadata

Get Azure Folder Metadata

ForEach

ForEachAzureFile

Activities

Copy Azure Data

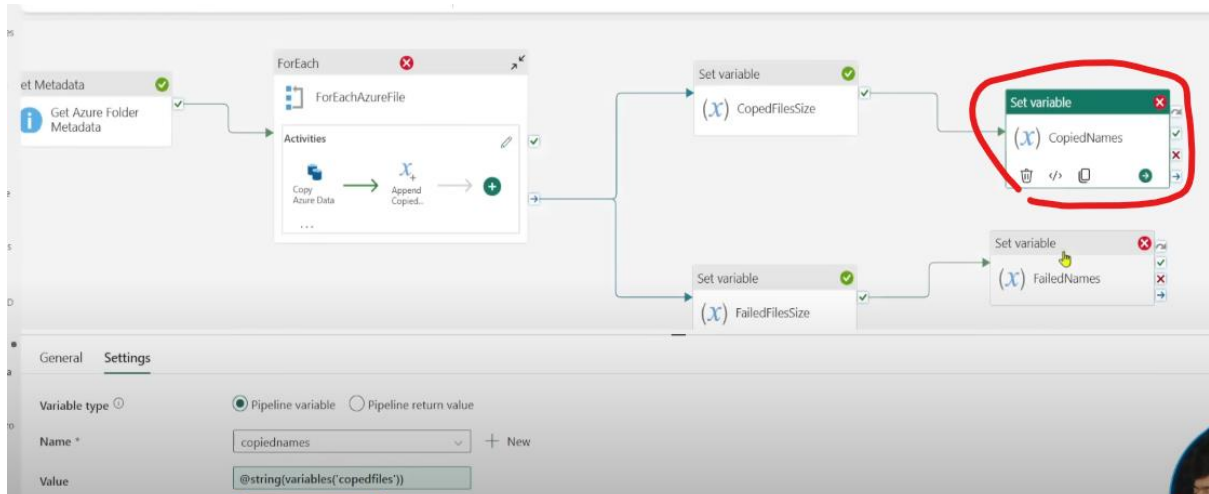
Append Copied...

Set variable (x) CopiedFilesSize

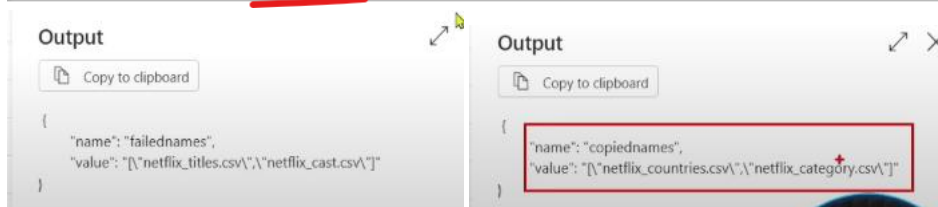
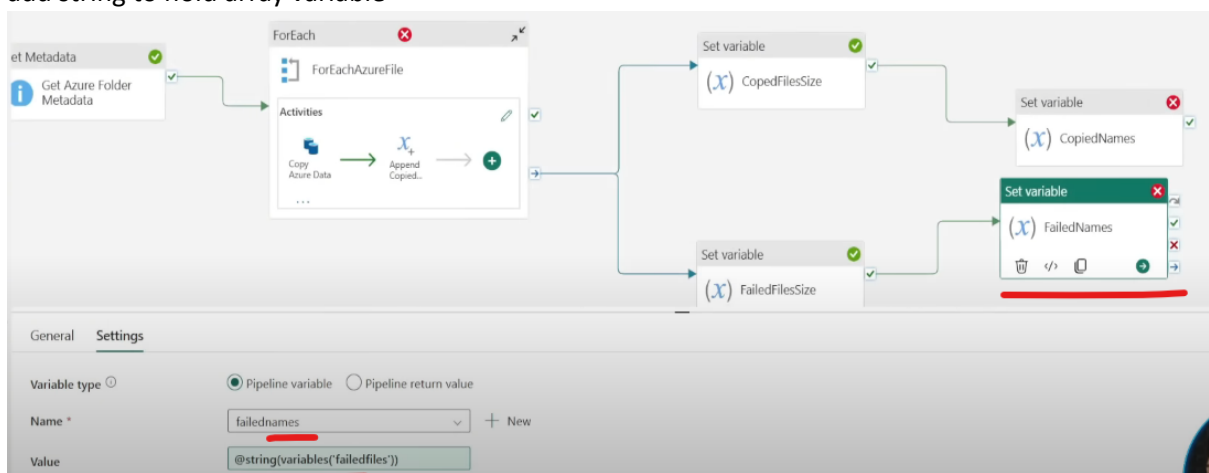
Set variable (x) CopiedNames

Set variable (x) FailedFilesSize

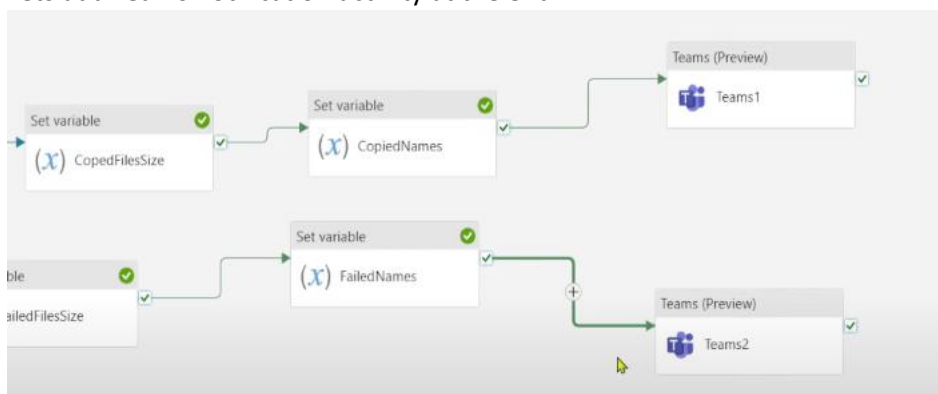
Set variable (x) FailedNames



add string to hold array variable



Lets add Teams notification activity at the end



Create a parent pipeline → use Invoke pipeline activity

General Settings

Currently Pipeline return value is only supported with ADF & Synapse pipelines. To fetch pipeline return value for Fabric pipelines, please use Invoke Pipeline (Legacy)

Type: ☒ Fabric ☐ Azure Data Factory ☐ Synapse

Connection * Refresh Edit

Workspace * Refresh

Pipeline * Refresh Open + New

Parameters

+ New Delete

Name	Type	Value
run_flag	String	Value

Treat as null

Wait on completion

create a pipeline parameter and feed it to child pipeline

Invoke Pipeline (Preview)

GitData

Parameters Variables Settings Output

+ New Delete

Name	Type	Default value
run_flag	String	yes

General Settings

Currently Pipeline return value is only supported with ADF & Synapse pipelines. To fetch pipeline return value for Fabric pipelines, please use Invoke Pipeline (Legacy)

Type: ☒ Fabric ☐ Azure Data Factory ☐ Synapse

Connection * Refresh Edit

Workspace * Refresh

Pipeline * Refresh Open + New

Parameters

+ New Delete

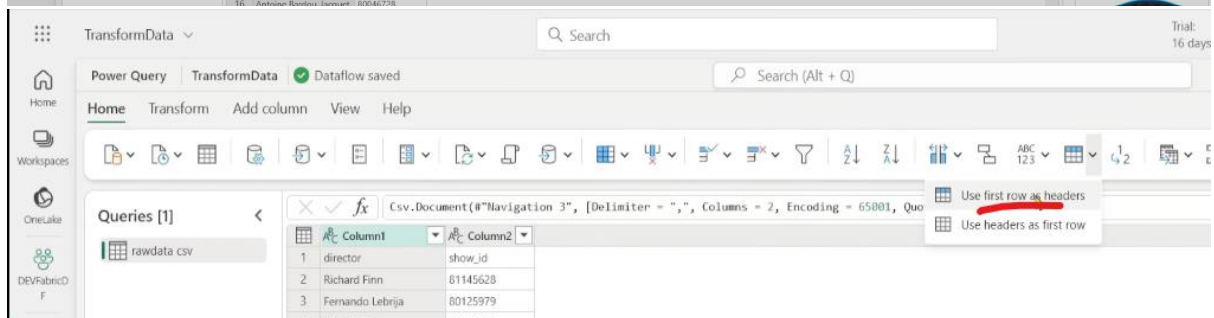
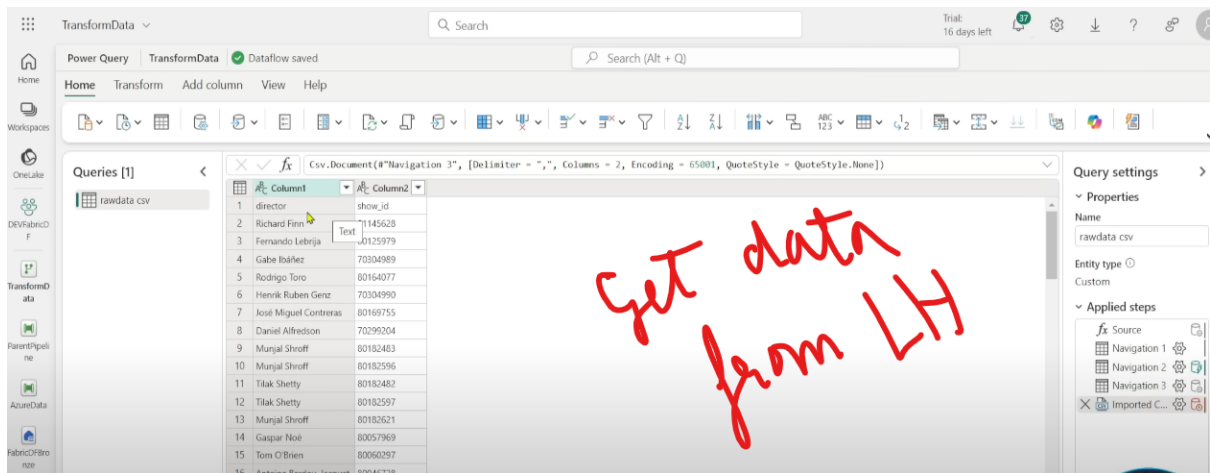
Name	Type	Value
run_flag	String	@pipeline().parameters.run_flag

Transformation

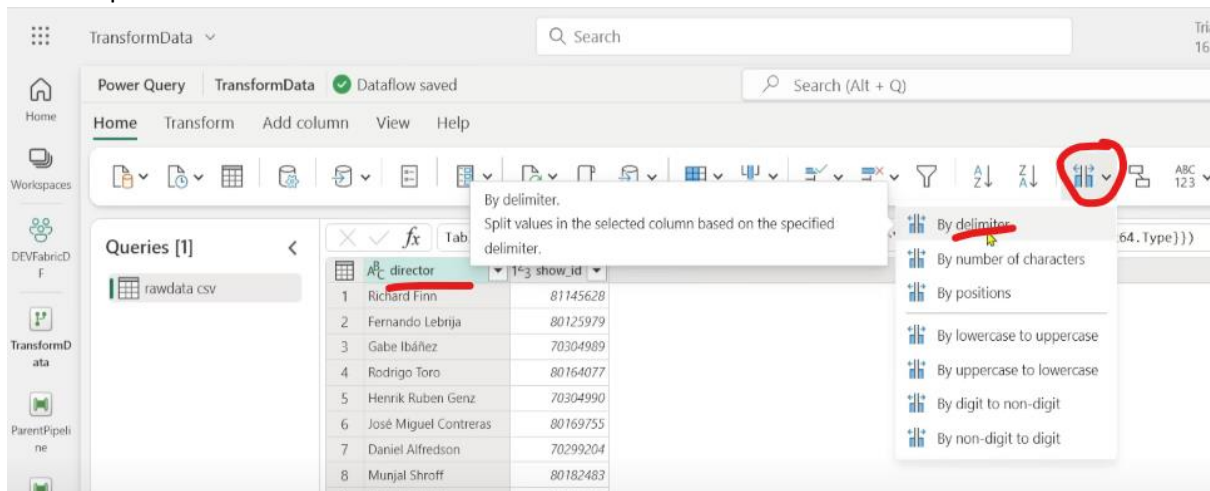
Data Flow Gen1 (backdated, use Gen2)

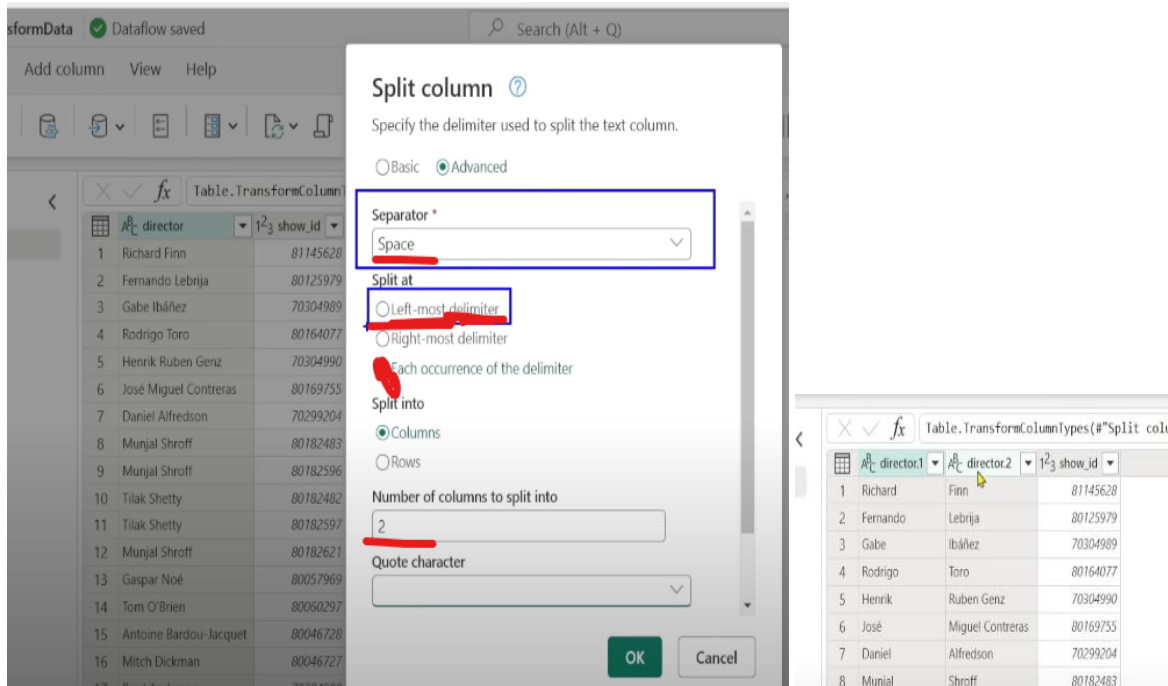
Create Data Flow Gen2 (Power Query online integrated with Fabric)→ Low code/no code manner

You can perform ETL in DFG2 but not ELT, but in Data Pipelines you can do ELT.

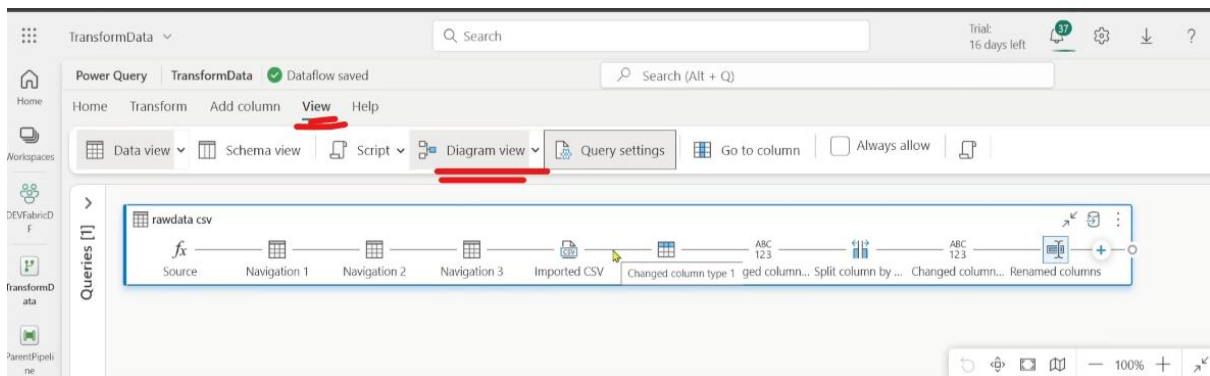


Wanna split the directors name

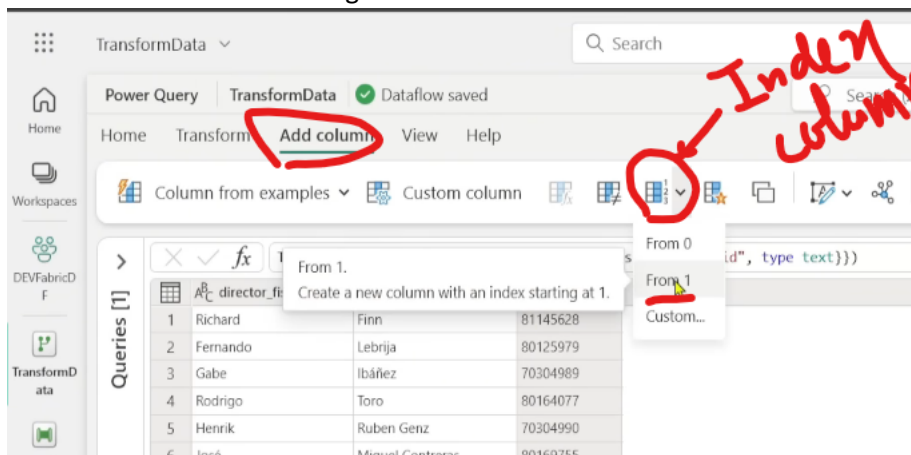




Rename the headers



Add a new column containing row number



TransformData

Power Query TransformData Dataflow saved

Home Transform Add column View Help

Column from examples Custom column

Table.AddIndexColumn(#"Changed column type 2", "Index", 1, 1, Int64.Type)

	director_firstname	director_lastname	show_id	Index
1	Richard	Finn	81145628	1
2	Fernando	Lebrija	80125979	2
3	Gabe	Ibáñez	70304989	3
4	Rodrigo	Toro	80164077	4
5	Henrik	Ruben Genz	70304990	5
6	José	Miguel Contreras	80169755	6
7	Daniel	Alfredson	70299204	7
8	Munjál	Shroff	80182483	8

I want to see how many movies are done by the same director, Group by on directors

TransformData

Power Query TransformData Dataflow saved

Home Transform Add column View Help

Group by.

Group rows in this table based on the values in the currently selected columns.

	director_firstname	director_lastname	show_id	Index
7	Daniel	Alfredson	70299204	7
8	Munjál	Shroff	80182483	8
9	Munjál	Shroff	80182596	9
10	Tilak	Shetty	80182482	10
11	Tilak	Shetty	80182597	11
12	Munjál	Shroff	80182621	12

Group by ?

Specify the column to group by and the desired output.

☒ Basic ☐ Advanced

Group by *

director_firstname

New column name *

Count

Operation *

Count rows

Column *

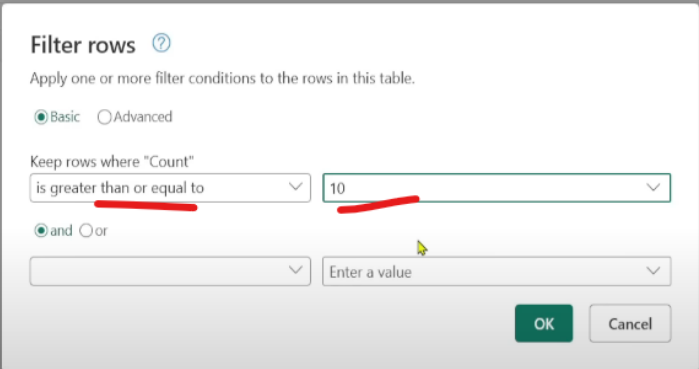
☐ Use fuzzy grouping

> Fuzzy group options

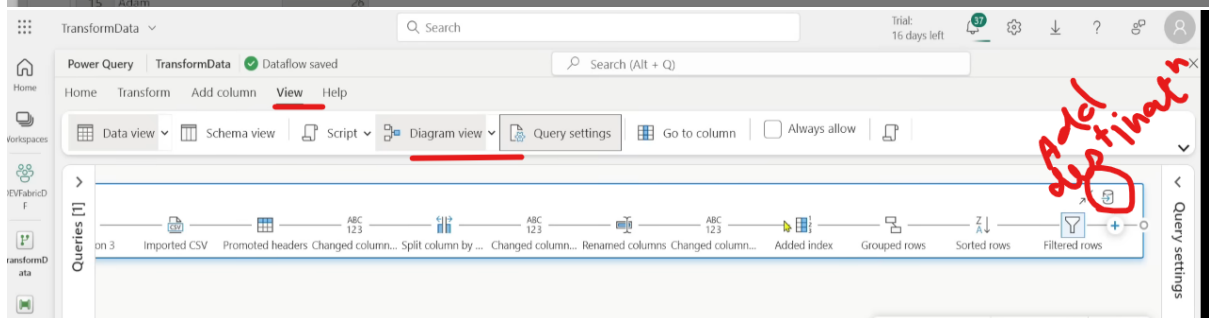
OK Cancel

	director_firstname	Count
1	Richard	24
2	Fernando	16
3	Gabe	3
4	Rodrigo	8
5	Henrik	2
6	José	4
7	Daniel	31
8	Munjial	3
9	Tilak	6
10	Gaspar	1
11	Tom	17

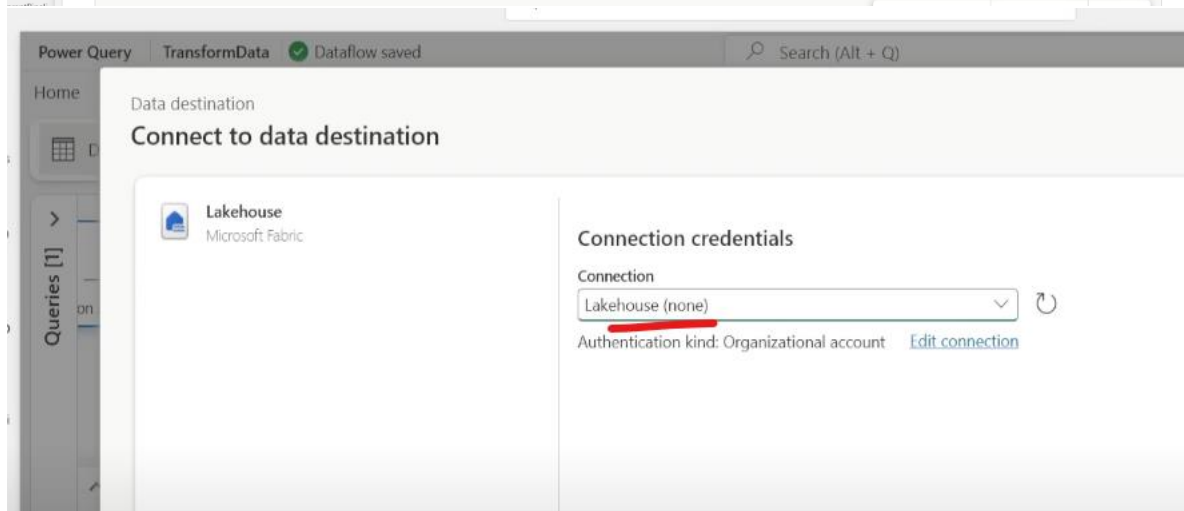
Filter only those if count more than 10



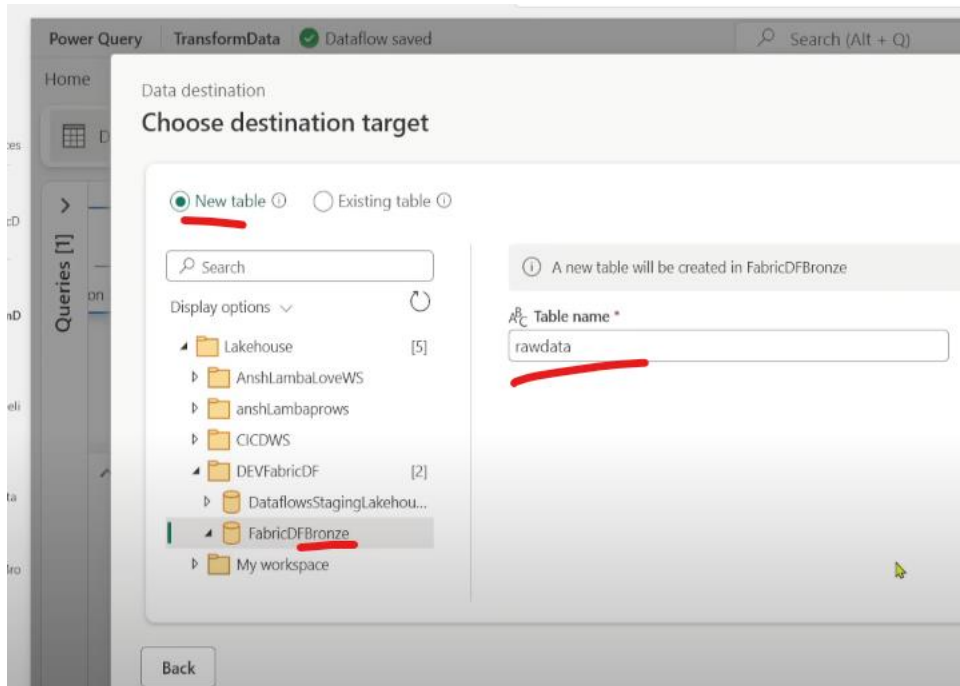
The image shows the Power Query 'Filter rows' dialog box. The 'Basic' tab is selected. The condition is set to 'Keep rows where "Count" is greater than or equal to 10'. The value '10' is entered in the text box. The 'OK' button is highlighted.



The image shows the Power Query 'View' tab with 'Diagram view' selected. The query steps are visible: Imported CSV, Promoted headers, Changed column..., Split column by..., Changed column..., Renamed columns, Changed column..., Added index, Grouped rows, Sorted rows, and Filtered rows. A red handwritten note 'And destination' is written over the 'Filtered rows' step.

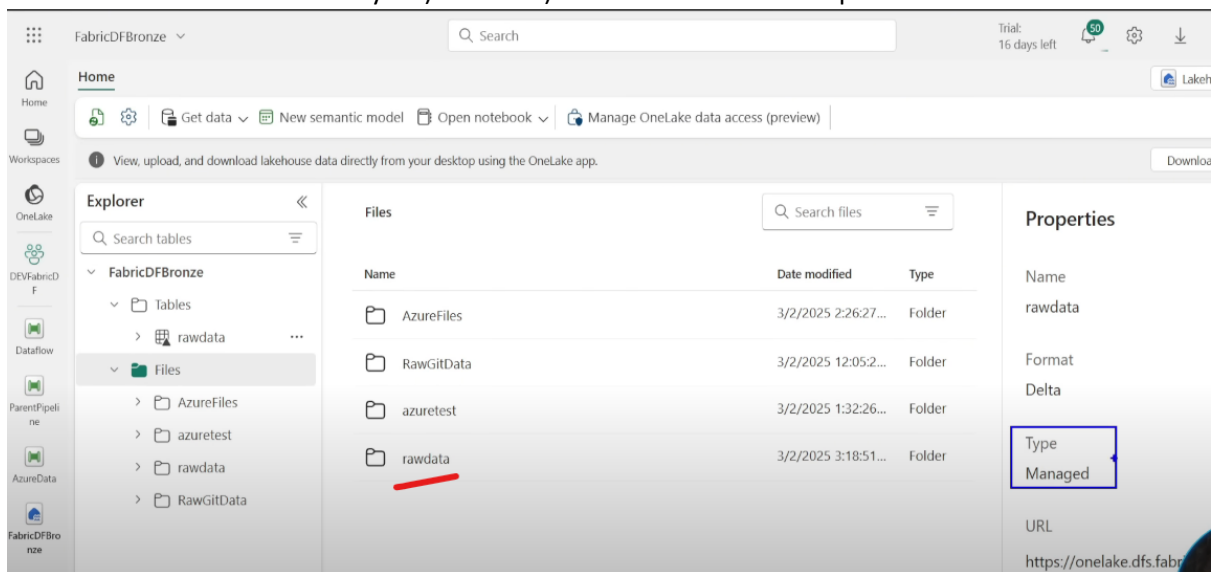


The image shows the Power Query 'Connect to data destination' dialog box. The 'Lakehouse' connection is selected. The 'Connection' dropdown is set to 'Lakehouse (none)'. The 'Authentication kind' is 'Organizational account'. The 'Edit connection' link is visible.



Publish the Data Flow, run it

We can run Data Flows in 2 ways: 1)manual 2)orchestrate with Data Pipeline



DEVFabricDF

Create deployment pipeline C

+ New item New folder → Import

	Name	Type	Task	Owner	Refreshed	Next refresh
	AzureData	Data pipeline	—	fabricde	—	—
	FabricDFBronze	Lakehouse	—	fabricde	—	—
	FabricDFBronze	Semantic m...	—	DEVFabricDF	2/3/2025, 12:...	N/A
	FabricDFBronze	SQL analyti...	—	fabricde	—	—
	FabricDFBronze_Con	Semantic m...	—	DEVFabricDF	2/3/2025, 12:...	N/A
	GitData	Data pipeline	—	fabricde	—	—
	ParentPipeline	Data pipeline	—	fabricde	—	—
	TransformData	Dataflow G...	—	fabricde	2/3/2025, ...	N/A

Refresh now

Data Transformation using Notebook

Home Edit Run View

Cancel all Standard session PySpark (Python) Environment Workspace default

Other people in your organization may have access to notebooks and Spark job definitions in this workspace

13	Gaspar Noé	80057969	Gaspar
14	Tom O'Brien	80060297	Tom
15	Antoine Bar...	80046728	Antoine
16	Mitch Dickm...	80046727	Mitch
17	Brad Anders...	70304988	Brad

```

1 df.write.format("csv")\
2   .mode("append")\
3   .saveAsTable("FabricDFBronze.rawdataspark")

```

[*] * 2 sec - Running

> Log

No triangle because it's a csv table managed

Use Notebook activity in Data Pipeline to orchestrate the flow

ParentPipeline

Home Activities Run View

Validate Run Schedule Add trigger (previ

Add Schedule Trigger to automate the flow

ParentPipeline

Search

Trial: 16 days left

HomeActivitiesRunView

Invoke Pipeline (Preview)

GitDataGitData

Validate

ParentPipeline

Data pipeline

AboutEndorsementSchedule

Not applicable (Previous schedule has expired. Update schedule and)

Run

Schedule

Scheduled run

OnOff

Repeat

Daily

Time

08:30 AM

Add a time

ParametersVariablesSettingsOutput

NewDelete

08:30 AM

Add a time

Start date and time

03/02/2025

End date and time

04/02/2025

Time zone

(UTC-04:00) Atlantic Time (Canada)

Apply

Discard

