

Azure Project Steps/Demo

Obj: Use Data Flow to move data to Azure Data Lake Storage or Azure Blob Storage. Use Databricks to read and write data from and to these data stores, and also perform advanced analytics and machine learning tasks.

The screenshot shows the 'New linked service' dialog in the Azure Data Factory portal. The dialog is for creating a new linked service of type 'Azure Data Lake Storage Gen2'. The 'Name' field is set to 'AzureDataLakeStorage1'. The 'Description' field is empty. The 'Connect via integration runtime' dropdown is set to 'AutoDetectIntegrationRuntime'. The 'Authentication type' dropdown is set to 'Account key'. The 'Account selection method' is set to 'From Azure subscription', and the 'Azure subscription' dropdown shows 'Azure subscription 1 (0b52c109-b0d1-4b02-9a0e-11d0c4b95081)'. The 'Storage account name' dropdown is set to 'mlgondy'. The 'Test connection' section has 'to linked service' selected. The 'Annotations' section has a '+ New' button. The 'Parameters' section has a '+ New' button. The 'Advanced' section has a '+' icon. The 'Set properties' panel on the right shows the 'Name' field set to 'monthly_revenue_original', the 'Linked service' dropdown set to 'AzureDataLakeStorage1', the 'File path' field set to '/Project/azure project / Month_value_1.csv', the 'First row as header' checkbox checked, and the 'Import schema' dropdown set to 'From connection/store'. The 'OK', 'Back', and 'Cancel' buttons are at the bottom.

New linked service

Name
AzureDataLakeStorage1

Description

Connect via integration runtime *
AutoDetectIntegrationRuntime

Authentication type
Account key

Account selection method
☒ From Azure subscription ☐ Enter manually

Azure subscription
Azure subscription 1 (0b52c109-b0d1-4b02-9a0e-11d0c4b95081)

Storage account name *
mlgondy

Test connection
☒ to linked service ☐ to file path

Annotations
+ New

Parameters
+ New

Advanced
+

Set properties

Name
monthly_revenue_original

Linked service *
AzureDataLakeStorage1

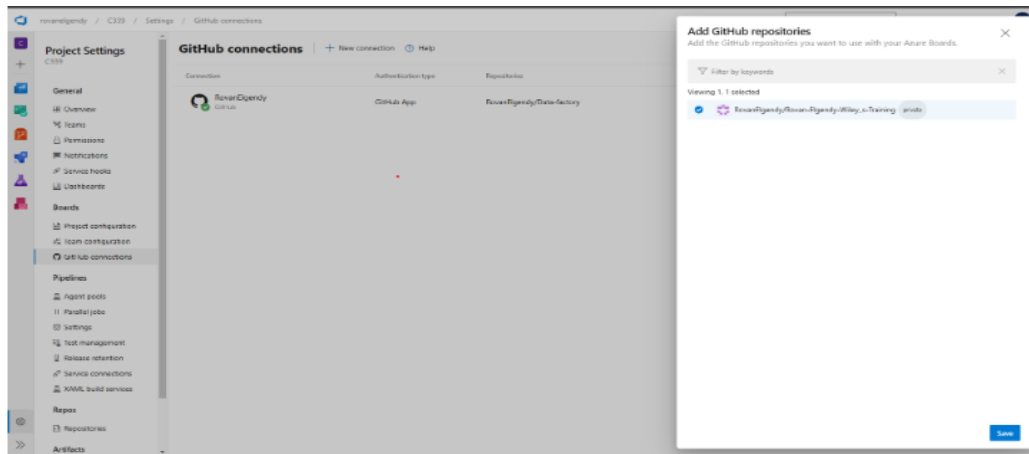
File path
historicaldata / Project/azure project / Month_value_1.csv

First row as header
☒

Import schema
☒ From connection/store ☐ From sample file ☐ None

OK **Back** **Cancel**

Setting up the Linked Services



Git configuration/integration to Git Repo
from organisation settings
after adding new Repo in Git

Source settings Source options **Projection** Optimize Inspect Data preview

Define default format Detect data type Import projection Reset schema

Column name	Type	Format
Period	date	dd.MM.yyyy
Revenue	float	Specify format
Sales_quantity	integer	Specify format
Average_cost	float	Specify format
The_average_annual_payroll_of_the_region	integer	Specify format

1- setting the data type for each column

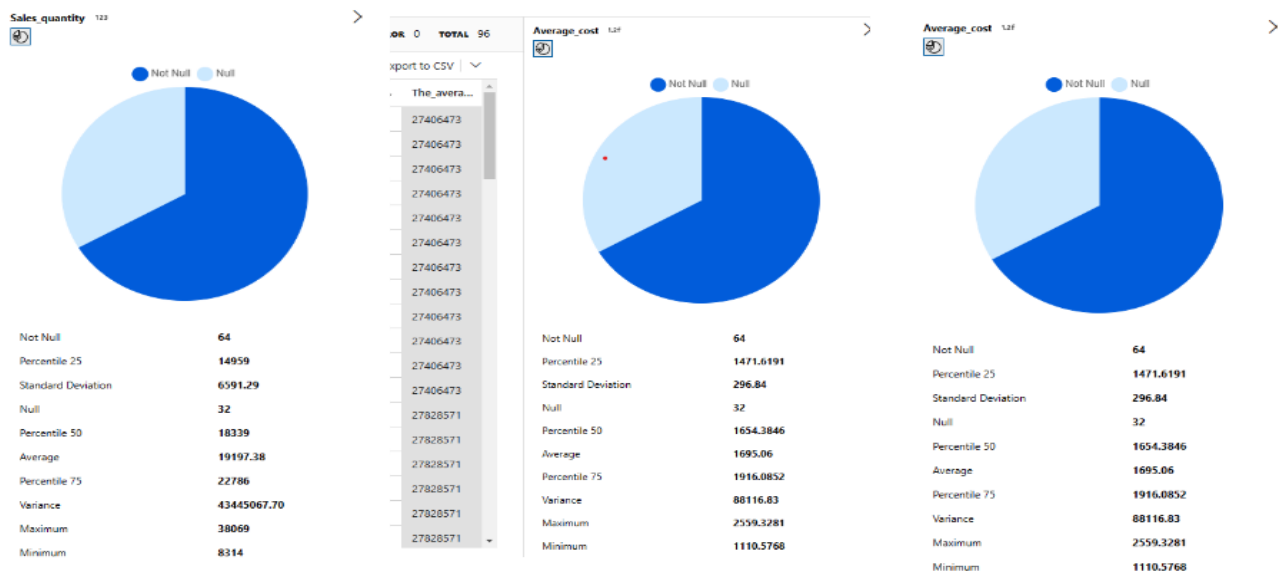
Source settings Source options Projection **Optimize** Inspect Data preview

Number of rows: INSERT 96 UPDATE 0 DELETE 0 UPSERT 0 LOOKUP 0 ERROR 0 TOTAL 96

Refresh Typecast Modify Map drifted Statistics Remove Export to CSV

Period	Revenue	Sales_quantity	Average_cost	The_average_annua...
2015-01-01	1.6010072E7	12729	1257.7635	3002.4676
2015-02-01	1.5807587E7	11636	1358.507	3002.4676
2015-03-01	2.2047146E7	15922	1384.697	3002.4676
2015-04-01	1.8814584E7	15227	1285.6067	3002.4676
2015-05-01	1.402148E7	8620	1626.6217	3002.4676
2015-06-01	1.6783928E7	13160	1275.3745	3002.4676

2- Data preview



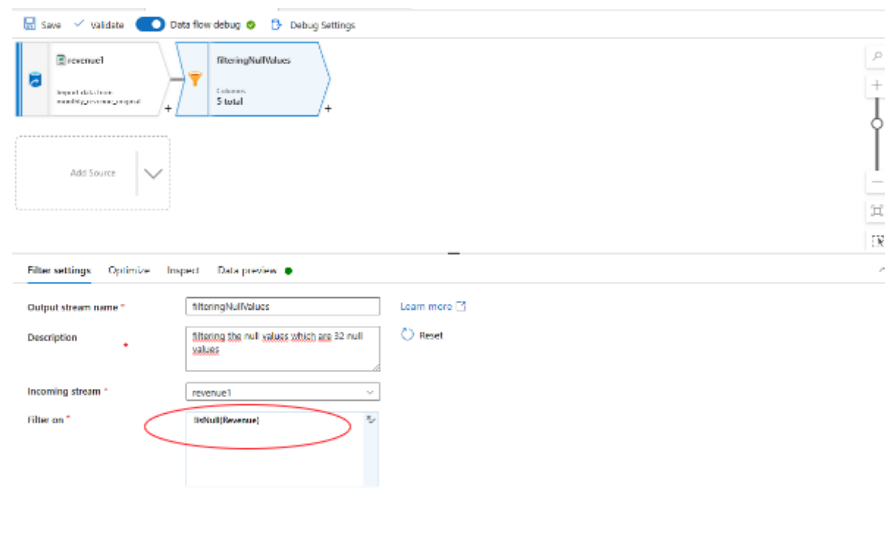
Revenue 1/27



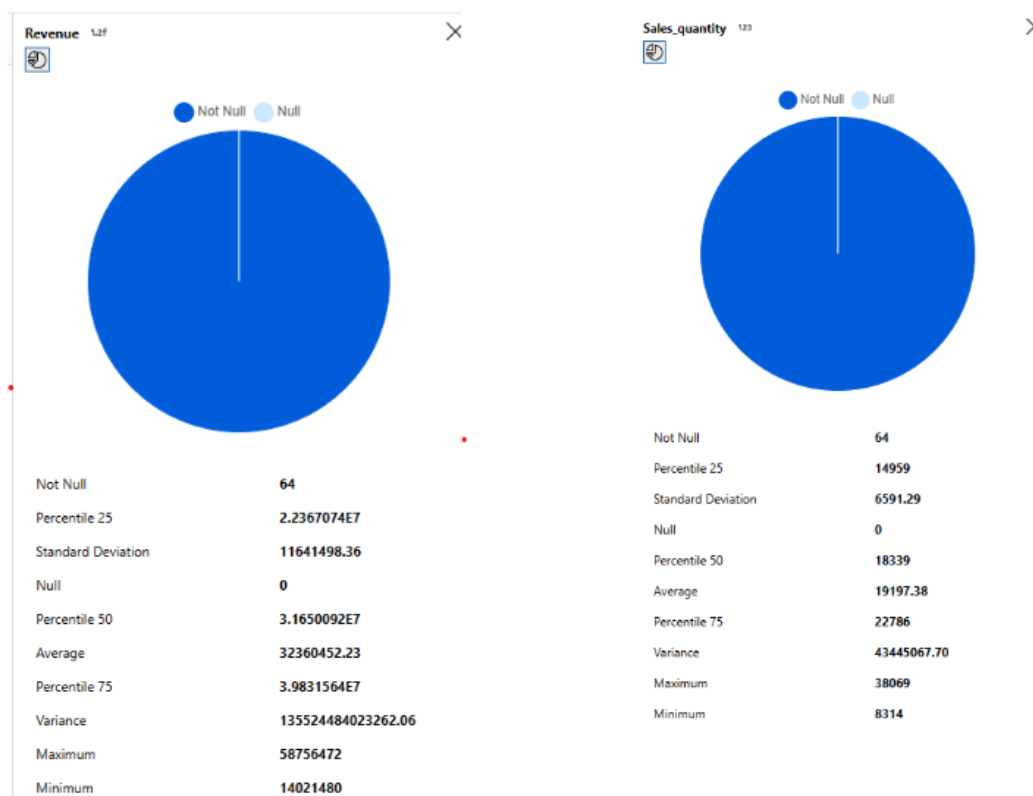
Not Null	64
Percentile 25	2.2367074E7
Standard Deviation	11641498.36
Null	32
Percentile 50	3.1650092E7
Average	32360452.23
Percentile 75	3.9831564E7
Variance	135524484023262.06
Maximum	58756472
Minimum	14021480

Data cleaning

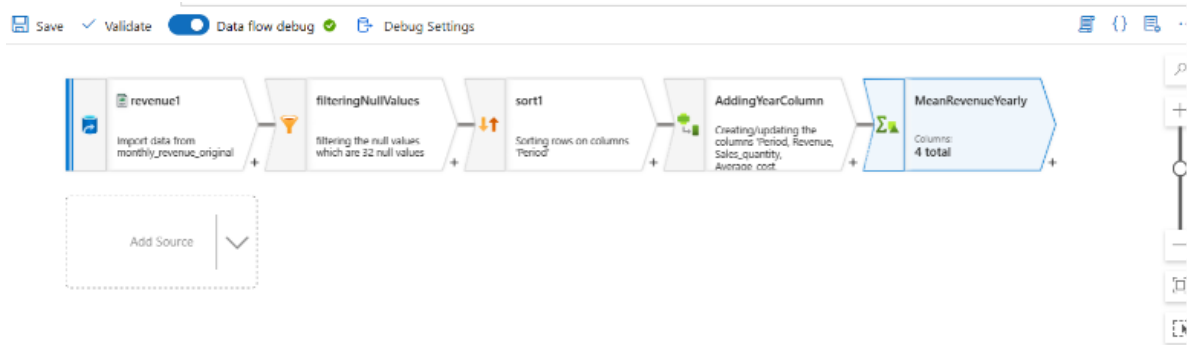
- Check the nan values from each column using the statistics above
- Using Filter data flow using the expression: `!isNull(Revenue)`



1- using Filter dataflow to remove the nan values using builder expression



2- Columns after filtering has zero Null values



Aggregate settings Optimize Inspect Data preview

Incoming stream * AddingYearColumn

Group by Aggregates

Columns Name as

Columns	Name as
123 Year	Year

+ -

Aggregate settings Optimize Inspect Data preview Previous Next

Output stream name * MeanRevenueYearly [Learn more](#)

Description getting the mean revenue yearly [Reset](#)

Incoming stream * AddingYearColumn

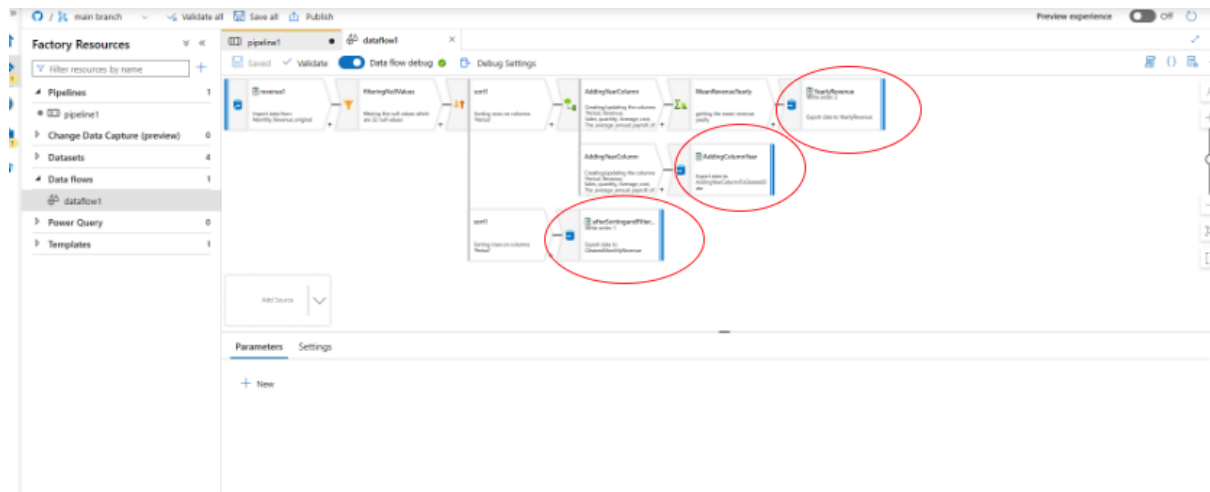
Group by Aggregates

Grouped by: Year

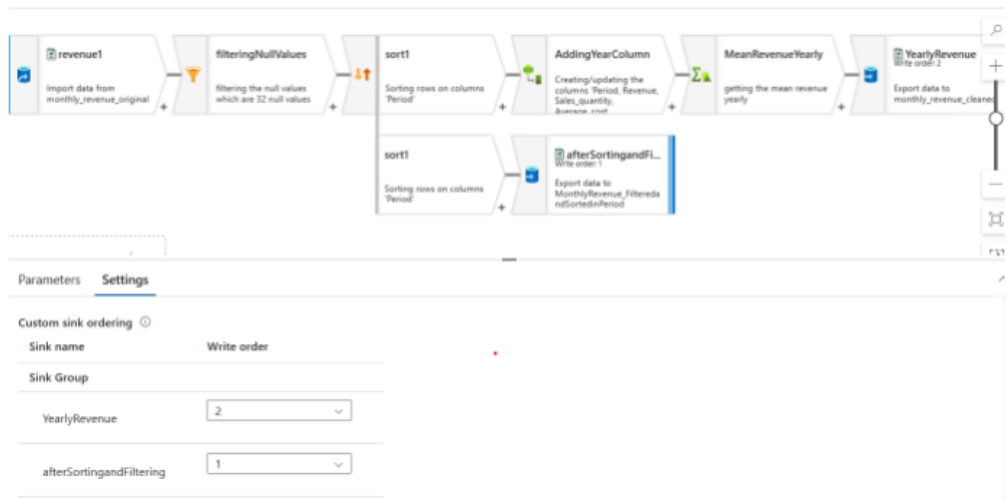
+ Add Clone Delete Open expression builder

Column	Expression
Mean_Revenue	toDecimal(avg(Revenue)) e^x + -
Mean_Sales_quantity	toDecimal(avg(Sales_quantity)) e^x + -
Mean_cost_of_Production	toDecimal(avg(Average_cost)) e^x + -

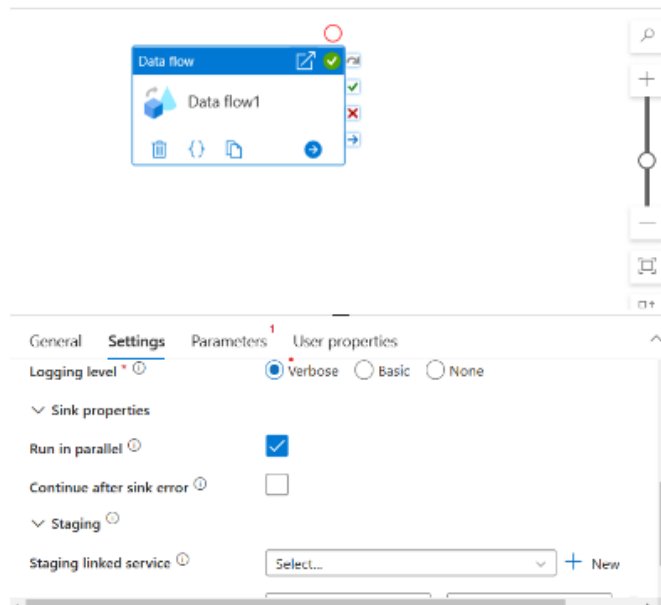
Getting the Avg of all columns per year using the aggregate transformation



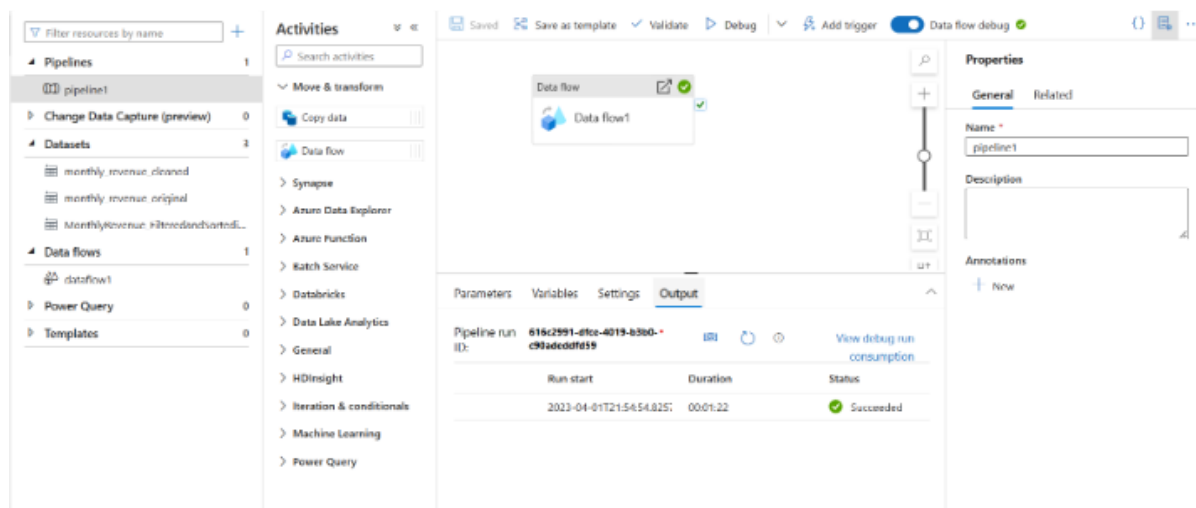
three sinks are added:
afterSortingandFiltering
AddingColumnYear
yearlyAvgRevenue



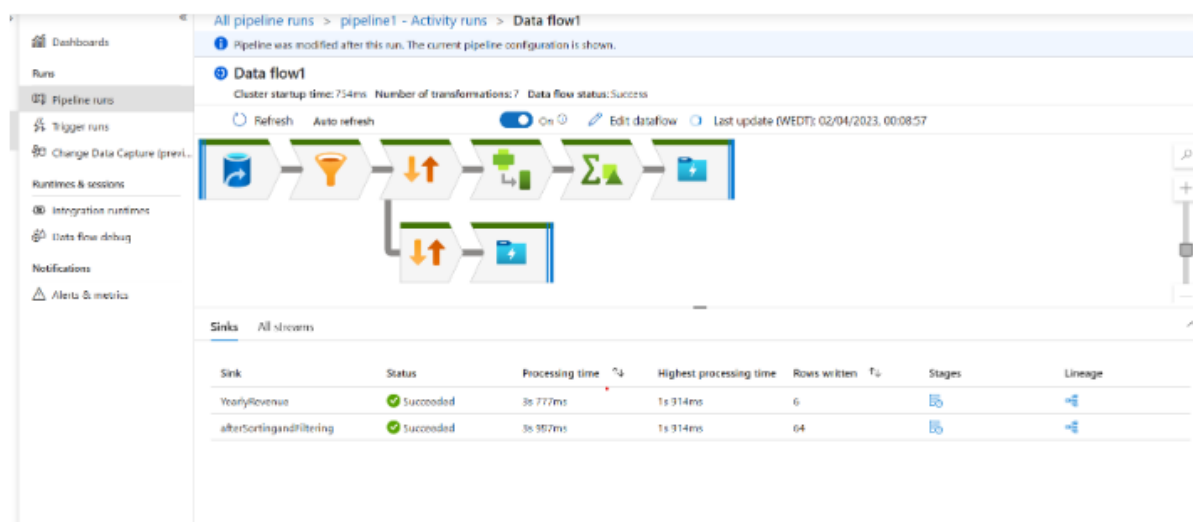
Writing the order of the three sinks



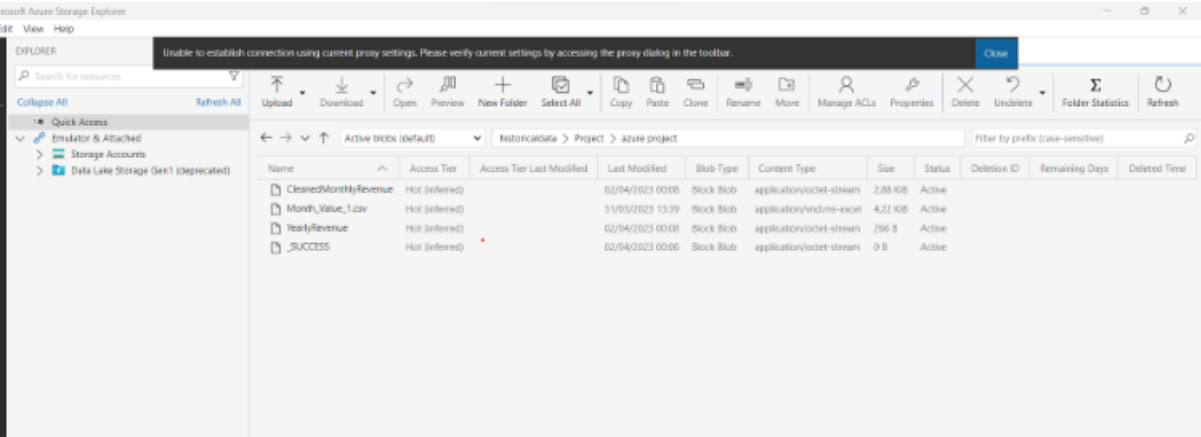
Creating new pipeline with our dataflow setting the sinks to run in parallel



Debugging the pipeline



Dataflow Debugging Dashboard after enabling the debug



the 3 Sinks have been added to the azure storage after debugging the pipeline

New trigger

Name *
UpdatingMonthRevenue

Description

Type *
Schedule

Start date * ⓘ
4/2/2023, 12:13:51 PM

Time zone * ⓘ
Coordinated Universal Time (UTC)

Recurrence * ⓘ
Every 1 Month(s)

Advanced recurrence options
☒ Month days ☐ Week days

Select day(s) of the month to execute

1	2	3	4	5	6	7
..

adding trigger to be triggered monthly

New trigger

☒ Month days ☐ Week days

Select day(s) of the month to execute

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	Last			

Execute at these times ⓘ

Hours

Minutes

Schedule execution times
12:13

☐ Specify an end date

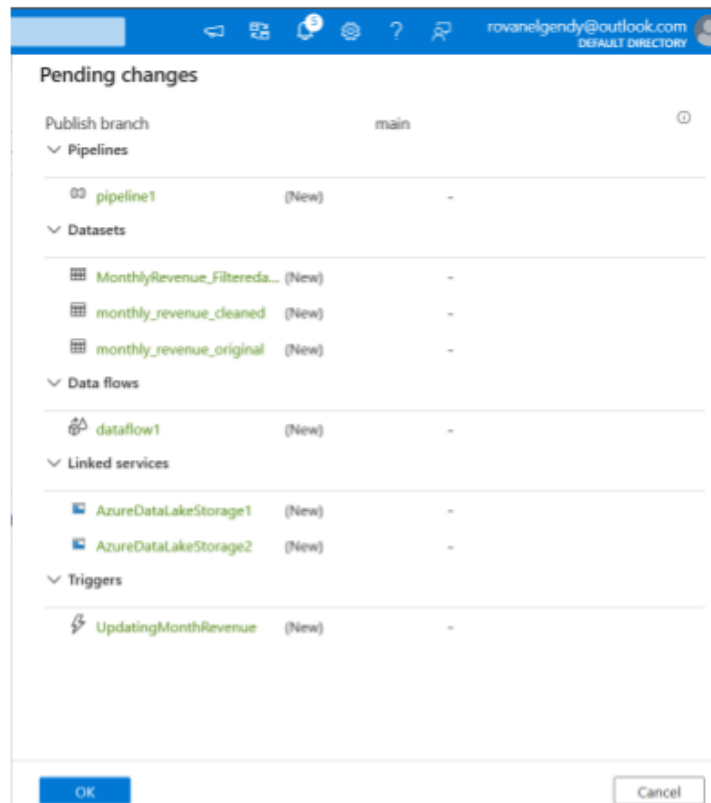
Annotations
+ New

Start trigger ⓘ
☒ Start trigger on creation

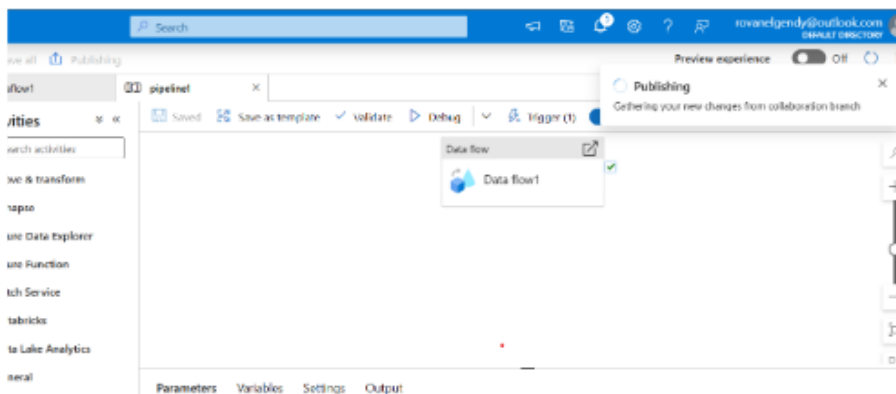
OK Cancel

Setting the trigger

Setting the trigger

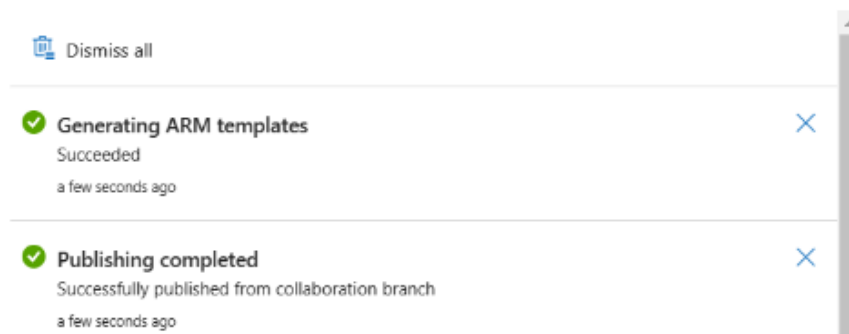


Saved and to be published



published

Notifications



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
55	01.01.2019	67.059011	6448.39938	1413.59552	64.267	198.0625																	
56	01.02.2019	67.071141	52398.2177	2398.48948	81.55	198.0625																	
57	01.03.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
58	01.04.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
59	01.05.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
60	01.06.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
61	01.07.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
62	01.08.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
63	01.09.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
64	01.10.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
65	01.11.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
66	01.12.2019	66.01645	1827.18437	75.552	8189.8994	78.0525																	
67	01.01.2020																						
68	01.02.2020																						
69	01.03.2020																						
70	01.04.2020																						
71	01.05.2020																						
72	01.06.2020																						
73	01.07.2020																						
74	01.08.2020																						
75	01.09.2020																						
76	01.10.2020																						
77	01.11.2020																						
78	01.12.2020																						
79	01.01.2021																						
80	01.02.2021																						

Updating the original dataset by one row

pipeline1 - Activity runs

Rerun updates

All run options have been condensed into a single button.

Activity runs

pipeline run id: 93b5e6d0c7d8-1c4b-v515-7716a67022a

All status: Export to CSV

Activity name	Status	Activity type	Run start	Duration	Log	Integration runtime
Data flow1	In progress	Data flow	4/2/2023, 2:58:04 PM	00:00:21		

pushing the trigger we made earlier manually

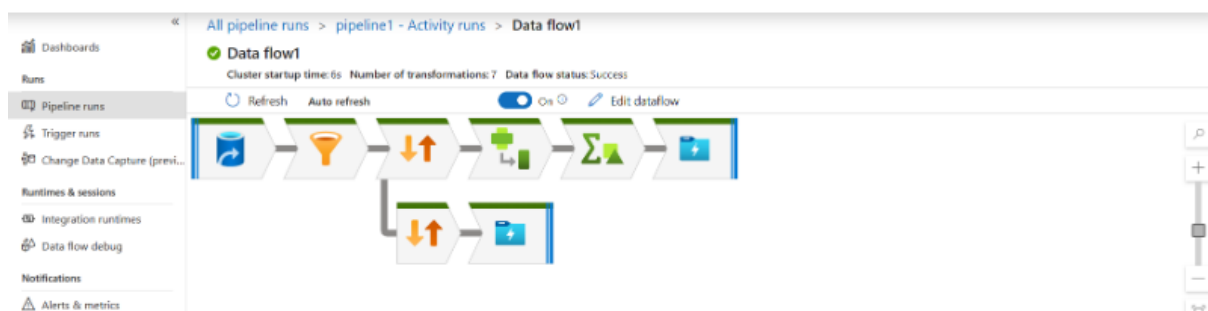
rovanelgenty@outlook.com
DEFAULT DIRECTORY

Notifications

Dismiss all

UpdatingMonthRevenue
UpdatingMonthRevenue has been saved.
a few seconds ago

Trigger success notification



Sinks All streams						
Sink	Status	Processing time	Highest processing time	Rows written	Stages	Lineage
YearlyRevenue	Succeeded	4s 189ms	2s 212ms	6		
afterSortingandFiltering	Succeeded	4s 439ms	2s 212ms	65		

here we can see after trigger the rows read is updated to 65 rows instead of 64 rows

pipeline1 dataflow1

Save Validate Data flow debug Debug Settings

revenue1 Import data from monthly_revenue_original

filteringNullValues filtering the null values which are 32 null values

sort1 Sorting rows on columns Period

AddingYearColumn Creating/updating the columns Period, Revenue, Sales_quantity, Average_cost

MeanRevenueYearly getting the mean revenue yearly

YearlyRevenue Write order:2 Export data to monthly_revenue_cleaned

sort1 Sorting rows on columns Period

afterSortingandFiltering Write order:1 Columns: 5 total

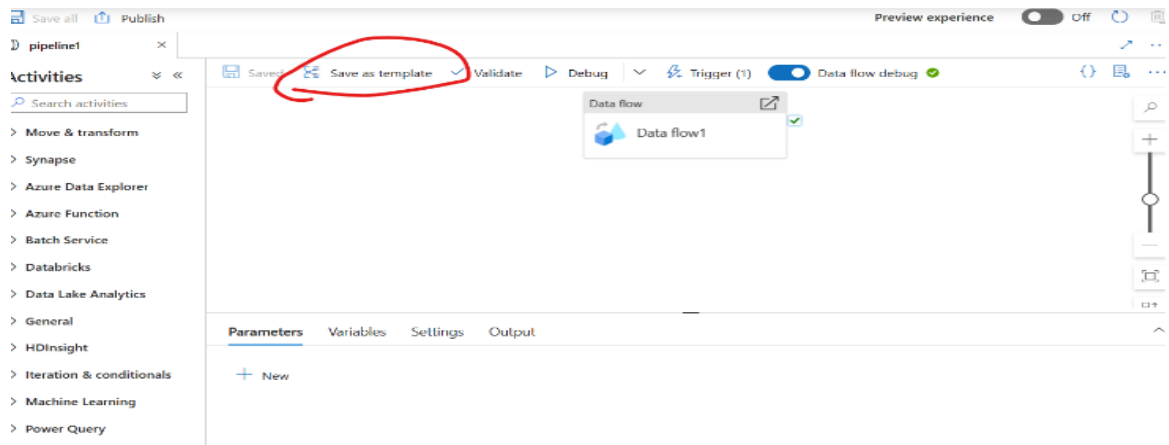
Sink Settings Errors Mapping Optimize Inspect Data preview

Number of rows + INSERT N/A + UPDATE N/A + DELETE N/A + UPSERT N/A LOOKUP N/A ERROR N/A TOTAL 65

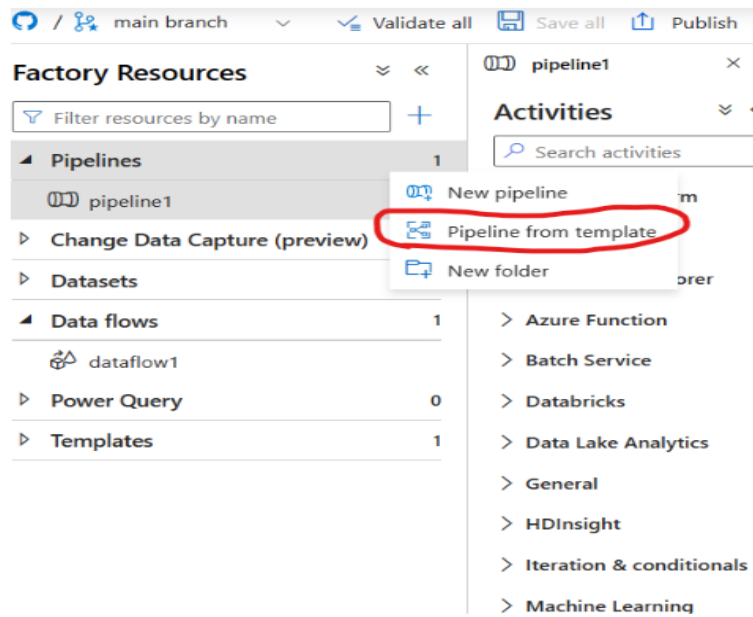
Refresh Statistics Export to CSV

Period	Revenue	Sales_quantity	Average_cost	The_average_annua...
2020-03-01	50022165	22369	1525.8827	29044998
2020-04-01	52320693	26615	1965.8348	29044998
2020-05-01	52320693	26615	1965.8348	29044998

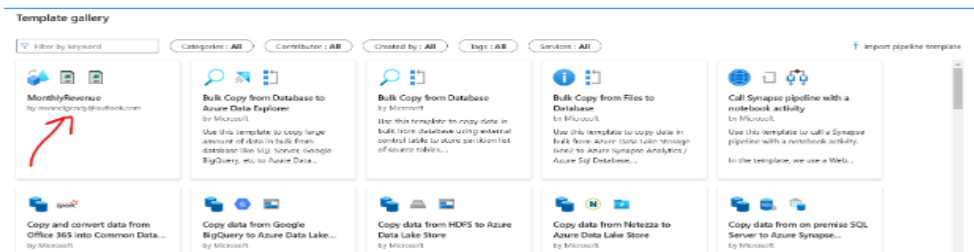
the updated row in the data preview



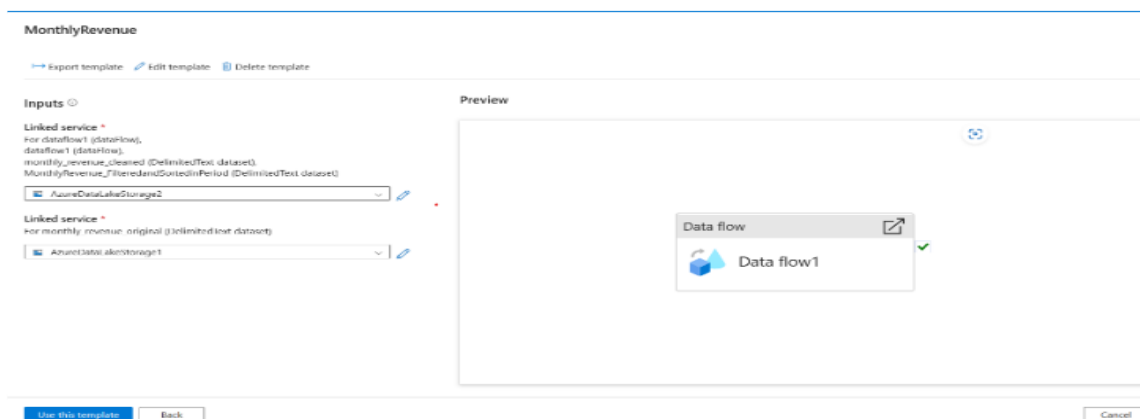
You can save the pipeline as template to import later



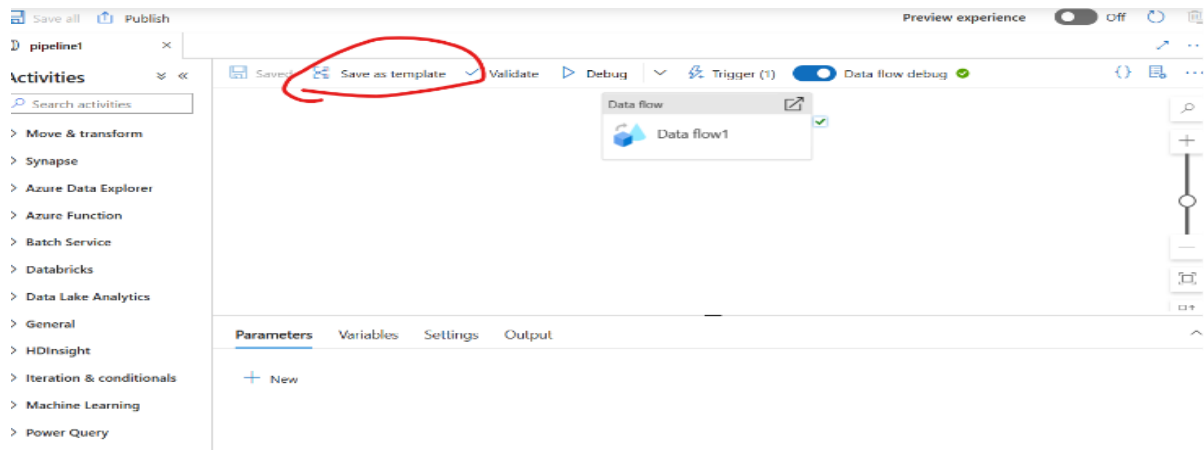
importing the template



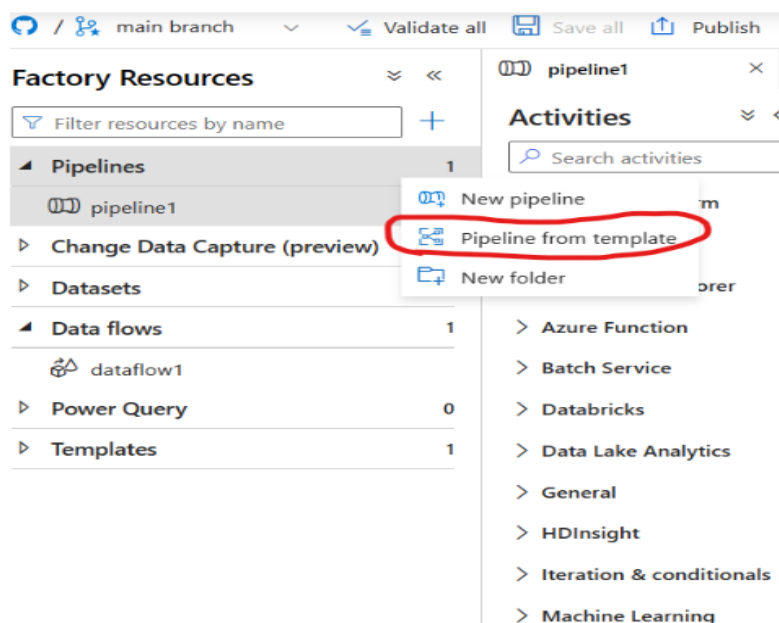
the template gallery



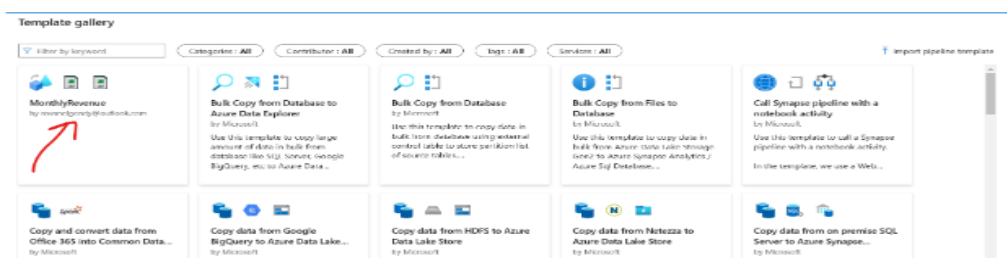
last step to click on use this template



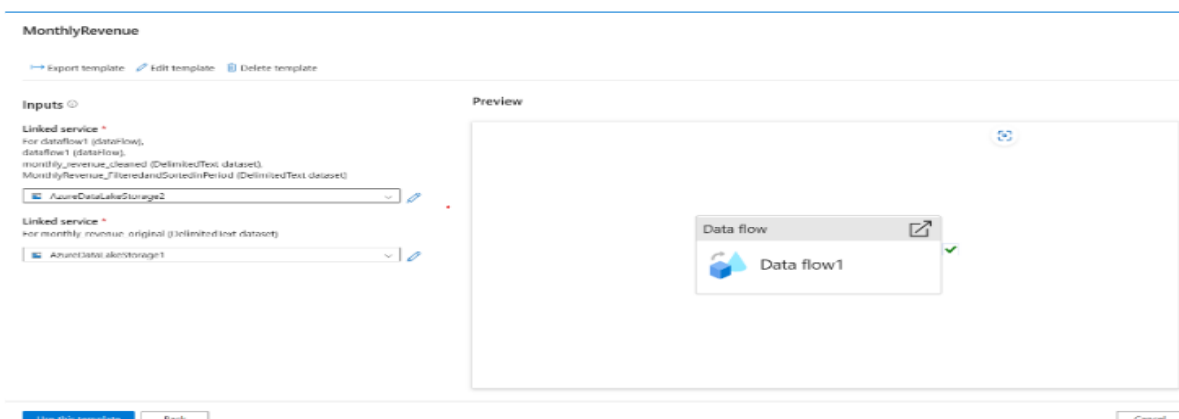
You can save the pipeline as template to import later



importing the template



the template gallery



last step to click on use this template

