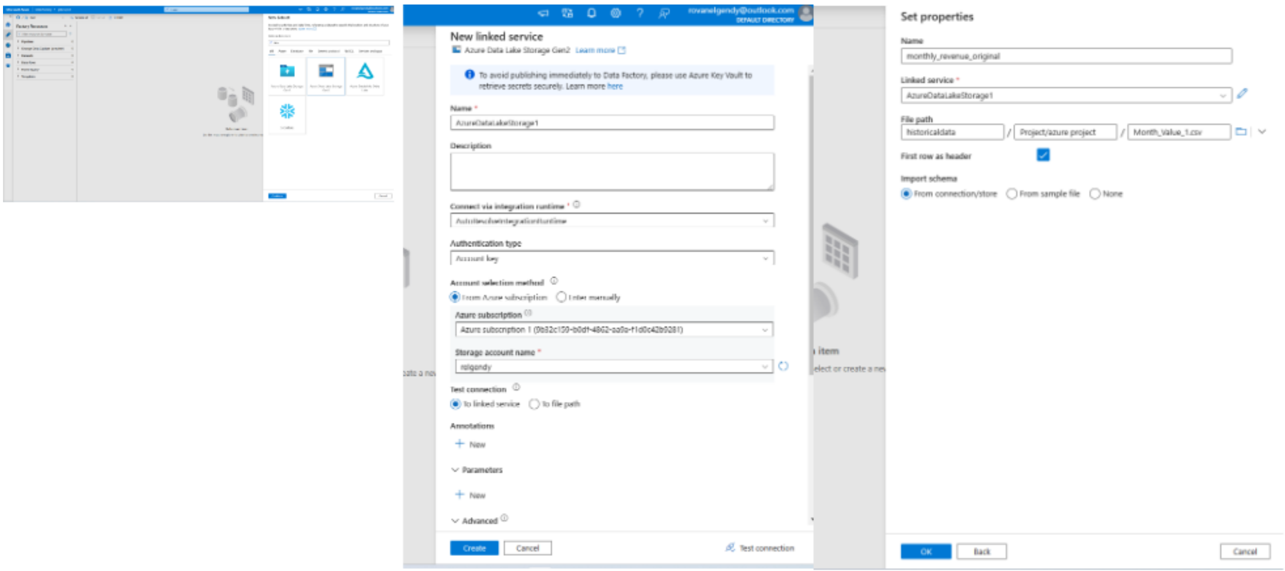
Azure Project Steps/Demo

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.



New linked service

Azure Data Lake Storage Gen2 Learn more

To avoid publishing immediately to Data Factory, please use Azure Key Vault to retrieve secrets securely. Learn more here

Name * AzureDataLakeStorage1

Description

Connect via integration runtime * AutoResolveIntegrationRuntime

Authentication type Account key

Account selection method ● From Azure subscription ○ Enter manually

Azure subscription Azure subscription 1 (9b32c159-b0df-4862-aa0a-f1d0c42b0281)

Storage account name * relgendy

Test connection ● To linked service ○ To file path

Annotations

Parameters

Advanced

Set properties

Name monthly_revenue_original

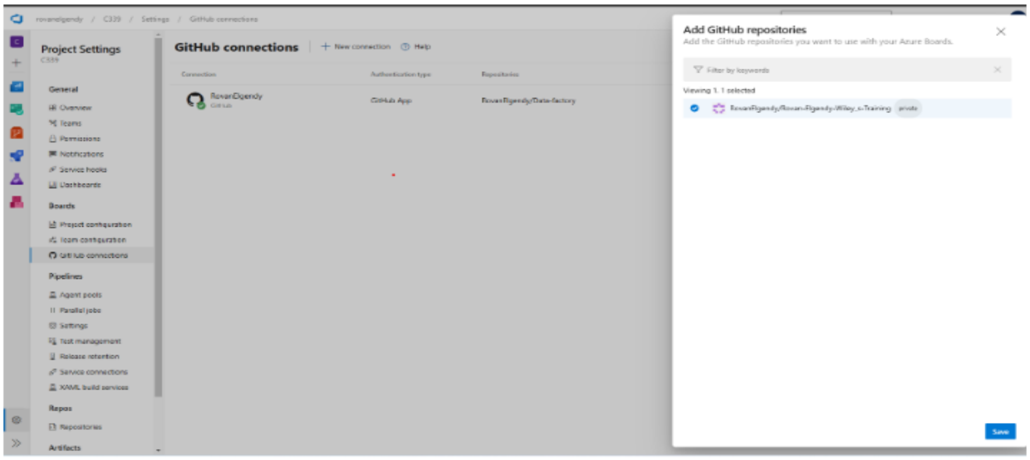
Linked service * AzureDataLakeStorage1

File path historicaldata / Project/azure project / Month_Value_1.csv

First row as header ☑

Import schema ● From connection/store ○ From sample file ○ None

Setting up the Linked Services



Connection	Authentication type	Repositories
RovanElgendy GitHub	GitHub App	RovanElgendy/Data-factory

Add GitHub repositories

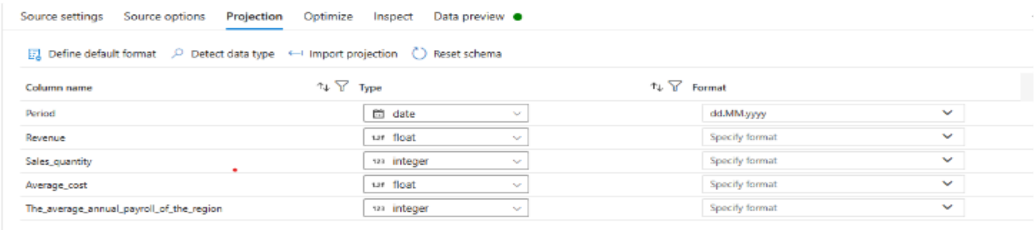
Add the GitHub repositories you want to use with your Azure Boards.

Filter by keywords

Viewing 1. 1 selected

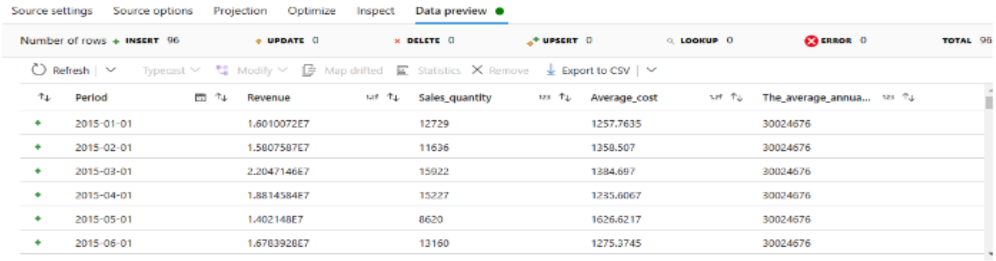
● RovanElgendy/Rovan-Elgendy-Wiley_s-Training private

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



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

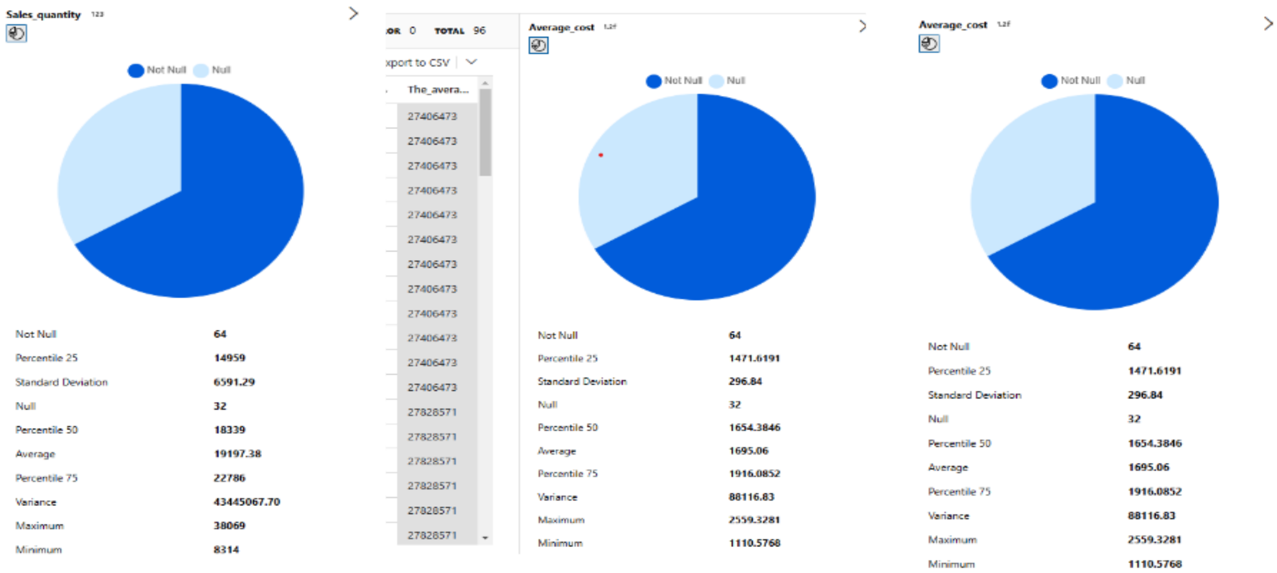
1- setting the data type for each column



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

	Period	Revenue	Sales_quantity	Average_cost	The_average_annua...
+	2015-01-01	1.6010072E7	12729	1257.7635	30024676
+	2015-02-01	1.5807587E7	11636	1358.507	30024676
+	2015-03-01	2.2047146E7	15922	1384.697	30024676
+	2015-04-01	1.8814584E7	15227	1235.6067	30024676
+	2015-05-01	1.402148E7	8620	1626.6217	30024676
+	2015-06-01	1.6783928E7	13160	1275.3745	30024676

2- Data preview



Sales_quantity

Not Null	64
Percentile 25	14959
Standard Deviation	6591.29
Null	32
Percentile 50	18339
Average	19197.38
Percentile 75	22786
Variance	43445067.70
Maximum	38069
Minimum	8314

The_avera...
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27406473
27828571
27828571
27828571
27828571
27828571
27828571

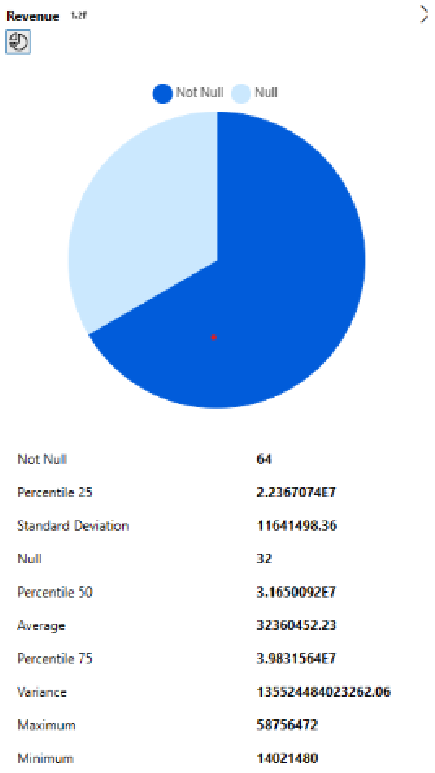
Average_cost

Not Null	64
Percentile 25	1471.6191
Standard Deviation	296.84
Null	32
Percentile 50	1654.3846
Average	1695.06
Percentile 75	1916.0852
Variance	88116.83
Maximum	2559.3281
Minimum	1110.5768

Average_cost

Not Null	64
Percentile 25	1471.6191
Standard Deviation	296.84
Null	32
Percentile 50	1654.3846
Average	1695.06
Percentile 75	1916.0852
Variance	88116.83
Maximum	2559.3281
Minimum	1110.5768

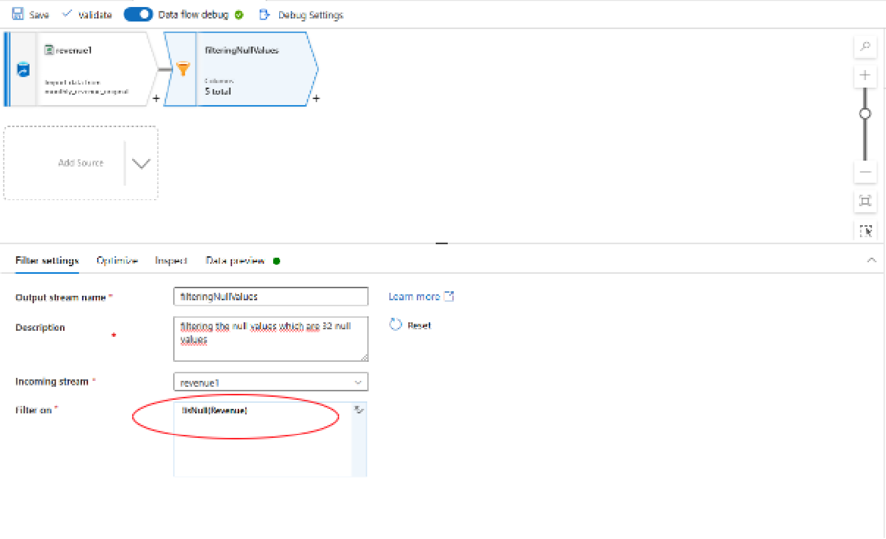
3- Dataset Description/Statistics



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)



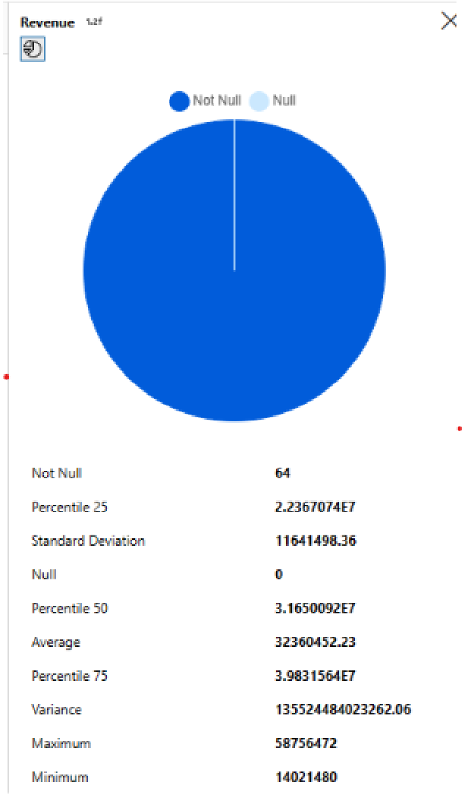
Output stream name * filteringNullValues

Description * filtering the null values which are 32 null values

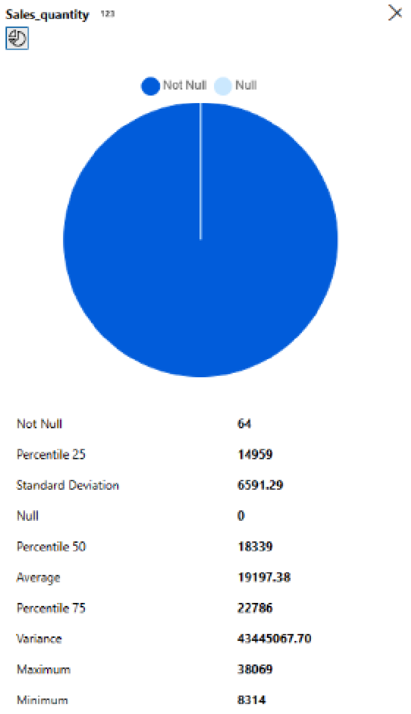
Incoming stream * revenue1

Filter on * !isNull(Revenue)

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

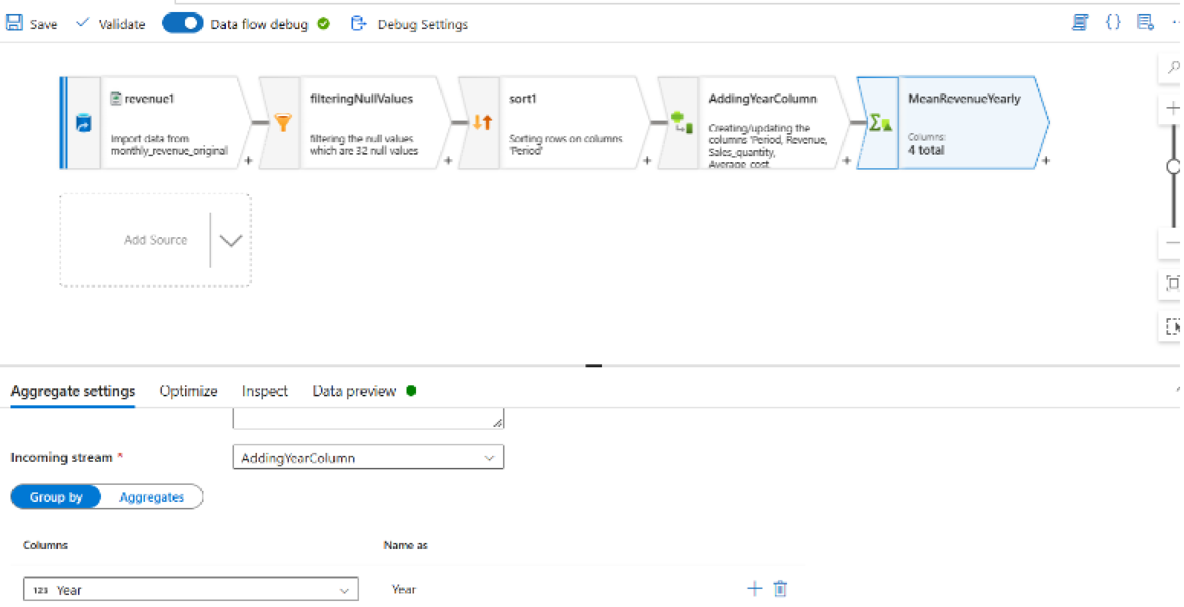


Not Null	64
Percentile 25	2.2367074E7
Standard Deviation	11641498.36
Null	0
Percentile 50	3.1650092E7
Average	32360452.23
Percentile 75	3.9831564E7
Variance	135524484023262.06
Maximum	58756472
Minimum	14021480



Not Null	64
Percentile 25	14959
Standard Deviation	6591.29
Null	0
Percentile 50	18339
Average	19197.38
Percentile 75	22786
Variance	43445067.70
Maximum	38069
Minimum	8314

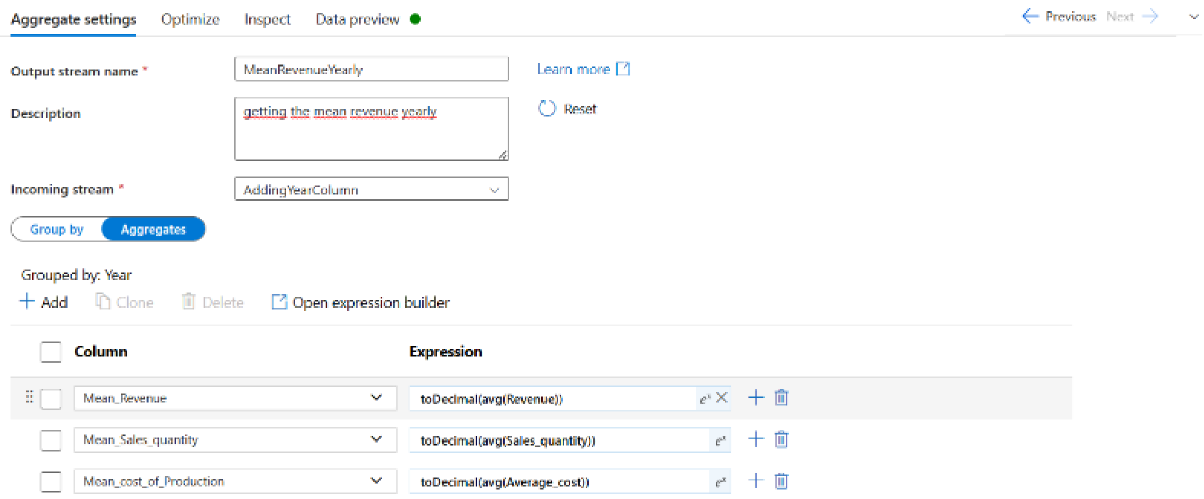
2- Columns after filtering has zero Null values


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
Columns:
4 total
Add Source
Aggregate settings
Optimize
Inspect
Data preview

Incoming stream * AddingYearColumn

Group by | Aggregates

Columns	Name as
123 Year	Year



Output stream name * MeanRevenueYearly

Description getting the mean revenue yearly

Incoming stream * AddingYearColumn

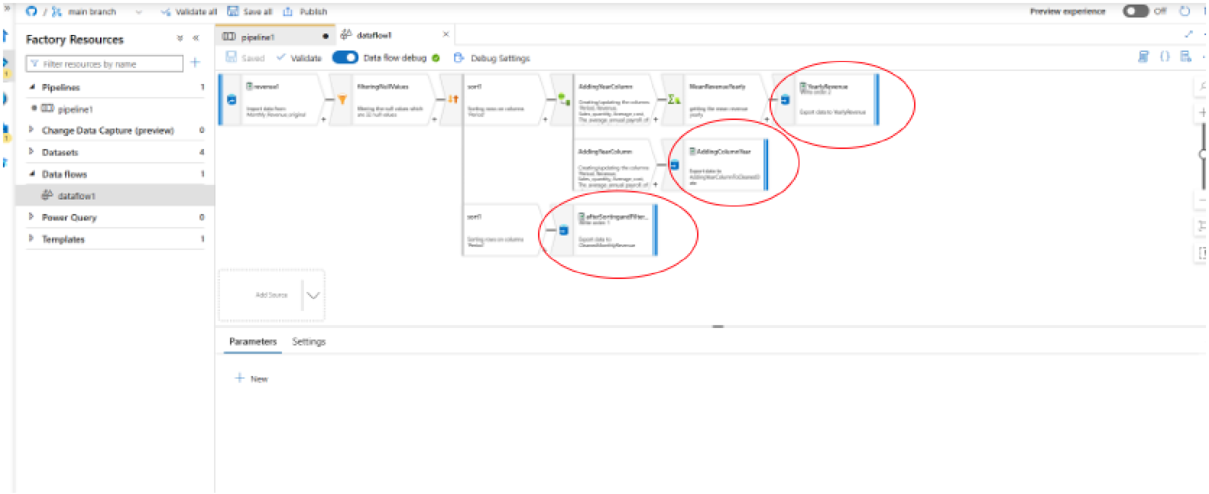
Group by | Aggregates

Grouped by: Year

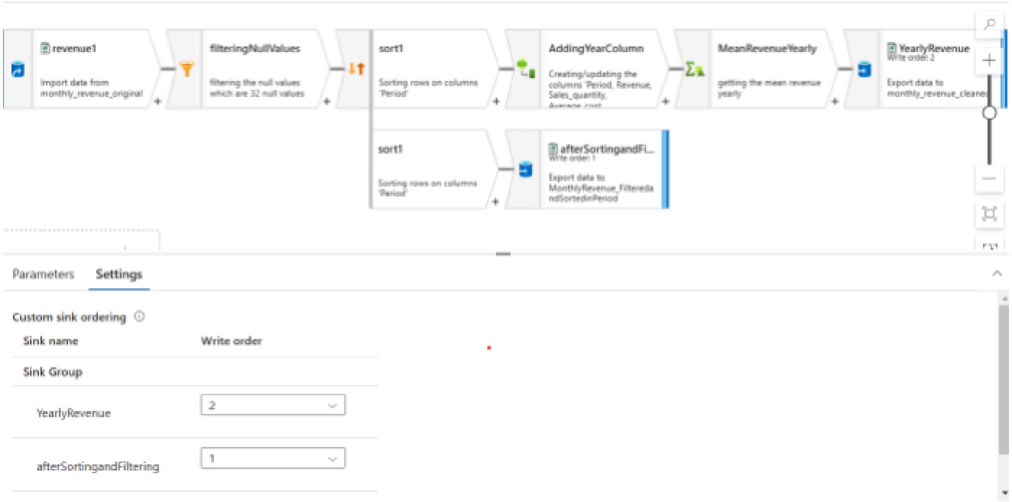
+ Add   Clone   Delete   Open expression builder

□ Column	Expression
□ Mean_Revenue	toDecimal(avg(Revenue))
□ Mean_Sales_quantity	toDecimal(avg(Sales_quantity))
□ Mean_cost_of_Production	toDecimal(avg(Average_cost))

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



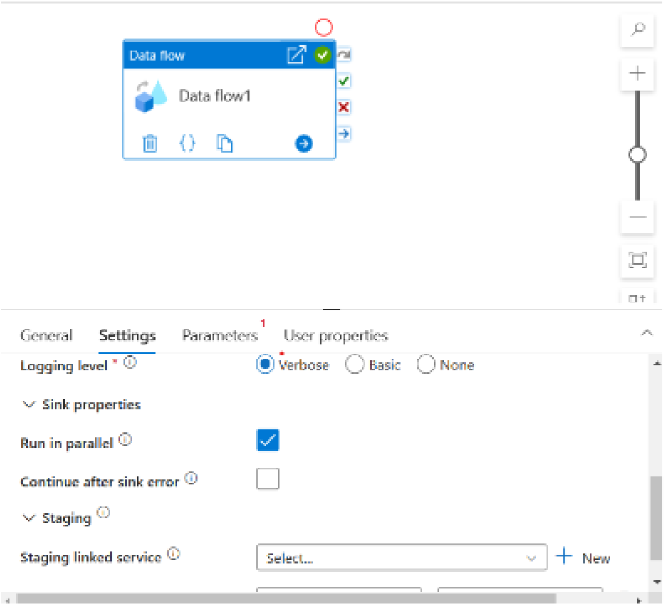
three sinks are added:
afterSortingandFiltering
AddingColumnYear
yearlyAvgRevenue


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'
afterSortingandFi...
Write order: 1
Export data to MonthlyRevenue_FilteredandSortedinPeriod
Parameters
Settings

Custom sink ordering

Sink name	Write order
Sink Group	
YearlyRevenue	2
afterSortingandFiltering	1

Writing the order of the three sinks



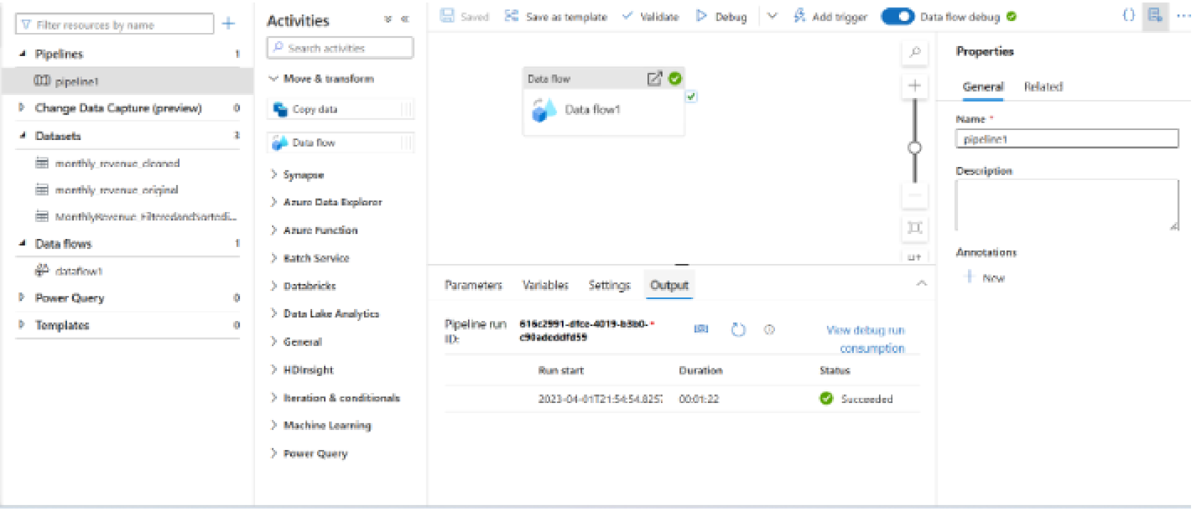
Logging level * ● Verbose ○ Basic ○ None

Run in parallel ☑

Continue after sink error □

Staging linked service: Select...

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



Name *: pipeline1

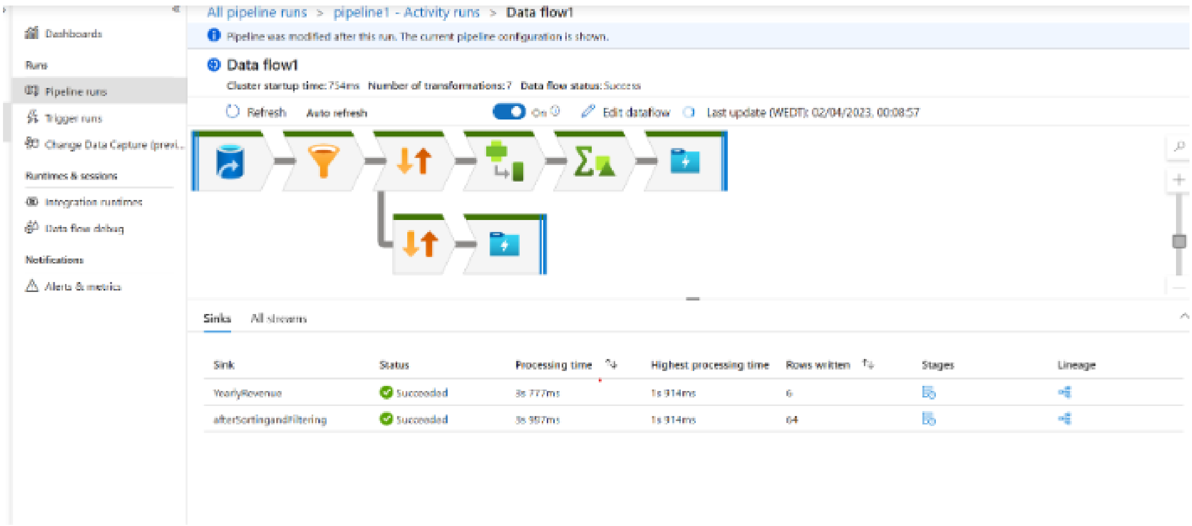
Description:

Pipeline run ID: 616c2991-dfce-4019-b5b0-c90adcddfd59

View debug run consumption

Run start	Duration	Status
2023-04-01T21:54:54.825	00:01:22	✓ Succeeded

Debugging the pipeline



All pipeline runs > pipeline1 - Activity runs > Data flow1

Pipeline was modified after this run. The current pipeline configuration is shown.

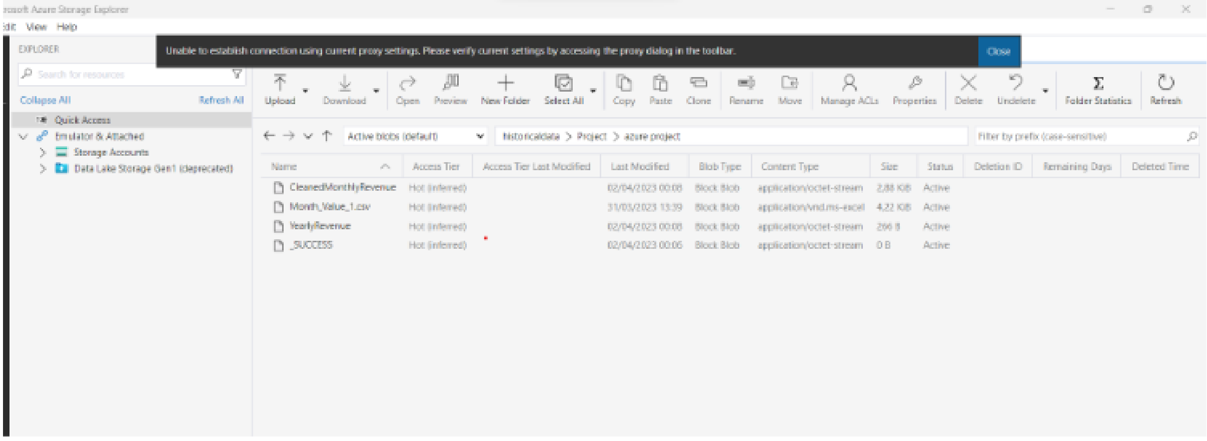
Data flow1

Cluster startup time: 754ms Number of transformations: 7 Data flow status: Success

Last update (WEDT): 02/04/2023, 00:08:57

Sink	Status	Processing time	Highest processing time	Rows written	Stages	Lineage
YearlyRevenue	✓ Succeeded	3s 777ms	1s 914ms	6		
afterSortingandFiltering	✓ Succeeded	3s 987ms	1s 914ms	64		

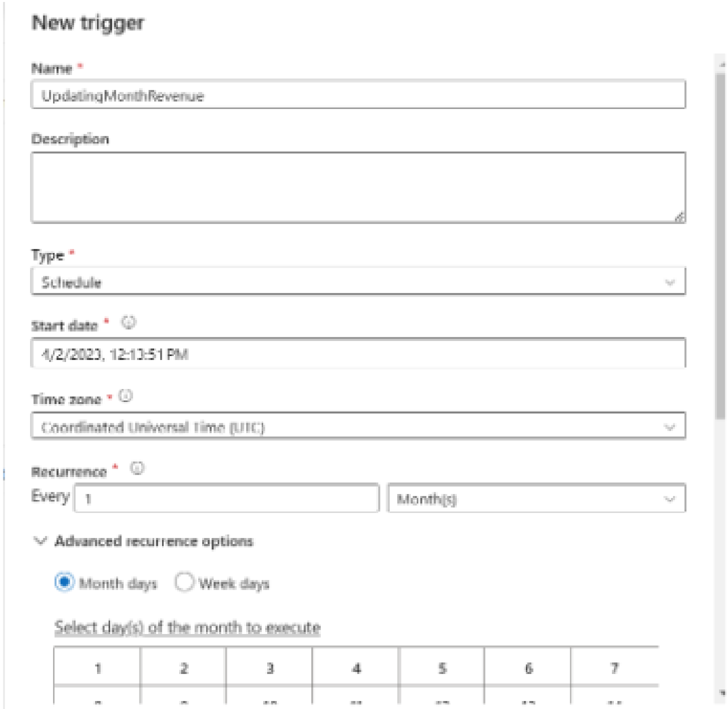
Dataflow Debugging Dashboard after enabling the debug



Unable to establish connection using current proxy settings. Please verify current settings by accessing the proxy dialog in the toolbar.

Name	Access Tier	Access Tier Last Modified	Last Modified	Blob Type	Content Type	Size	Status	Deletion ID	Remaining Days	Deleted Time
CleanedMonthlyRevenue	Hot (inferred)		02/04/2023 00:08	Block Blob	application/octet-stream	2,88 KiB	Active			
Month_Value_1.csv	Hot (inferred)		31/03/2023 13:39	Block Blob	application/vnd.ms-excel	4,22 KiB	Active			
YearlyRevenue	Hot (inferred)		02/04/2023 00:08	Block Blob	application/octet-stream	266 B	Active			
_SUCCESS	Hot (inferred)		02/04/2023 00:06	Block Blob	application/octet-stream	0 B	Active			

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)

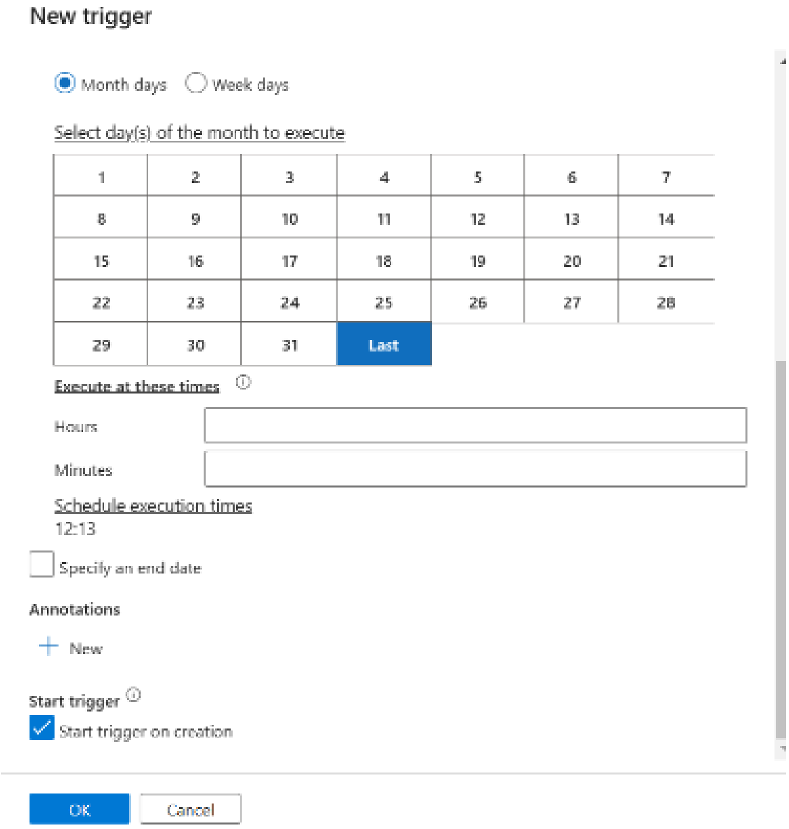
Advanced recurrence options

● 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

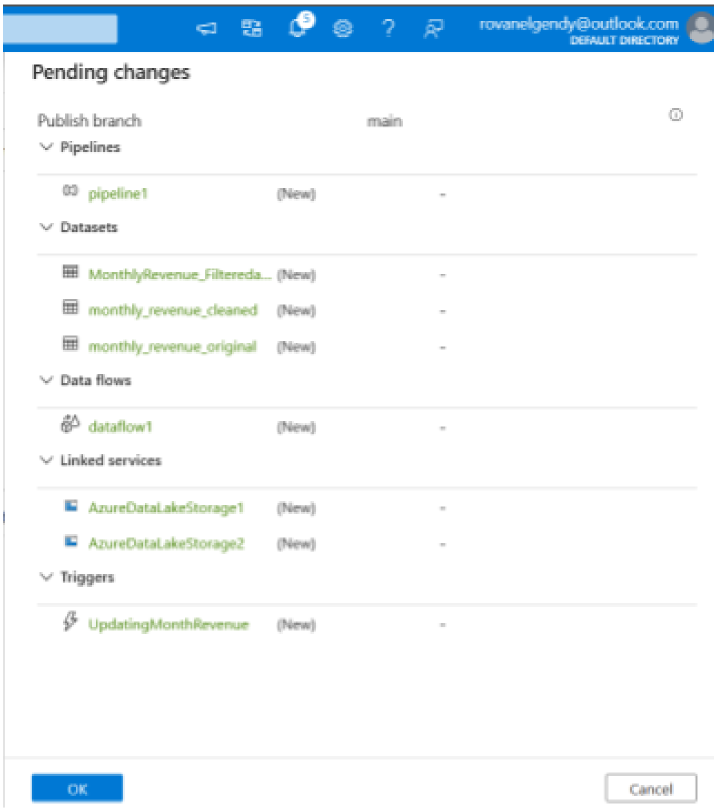
Annotations

Start trigger

☑ Start trigger on creation

Setting the trigger

Setting the trigger



rovanelgendy@outlook.com
DEFAULT DIRECTORY

Pending changes

Publish branch main

Pipelines

pipeline1	(New)	-

Datasets

MonthlyRevenue_Filtereda...	(New)	-
monthly_revenue_cleaned	(New)	-
monthly_revenue_original	(New)	-

Data flows

dataflow1	(New)	-

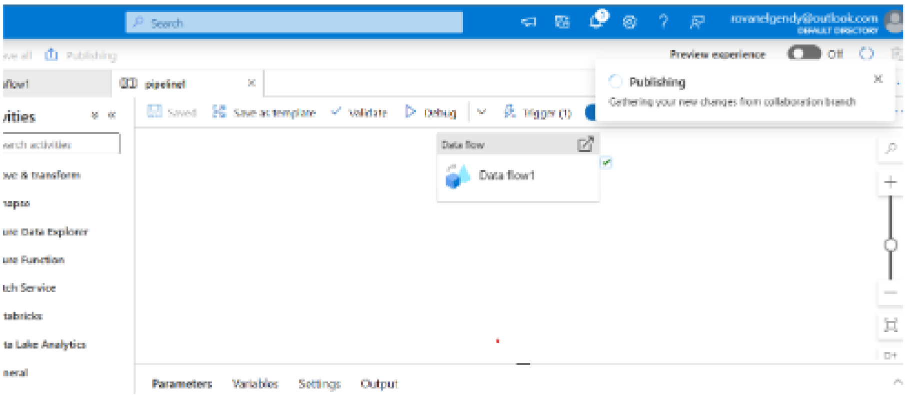
Linked services

AzureDataLakeStorage1	(New)	-
AzureDataLakeStorage2	(New)	-

Triggers

UpdatingMonthRevenue	(New)	-

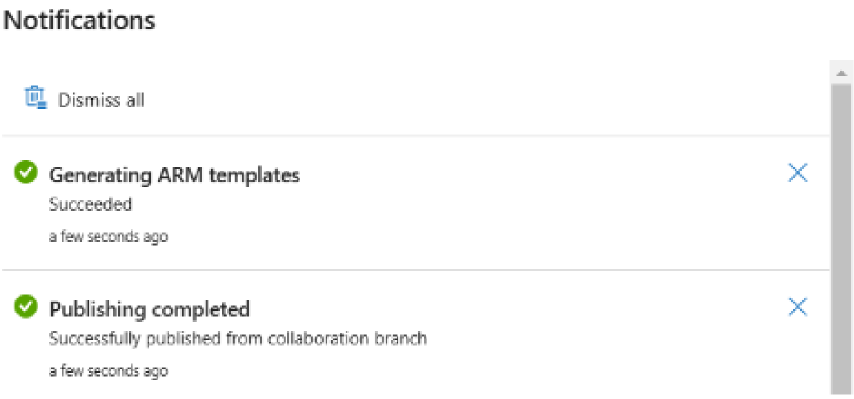
Saved and to be published



rovanelgendy@outlook.com
DEFAULT DIRECTORY

Publishing
Gathering your new changes from collaboration branch

published

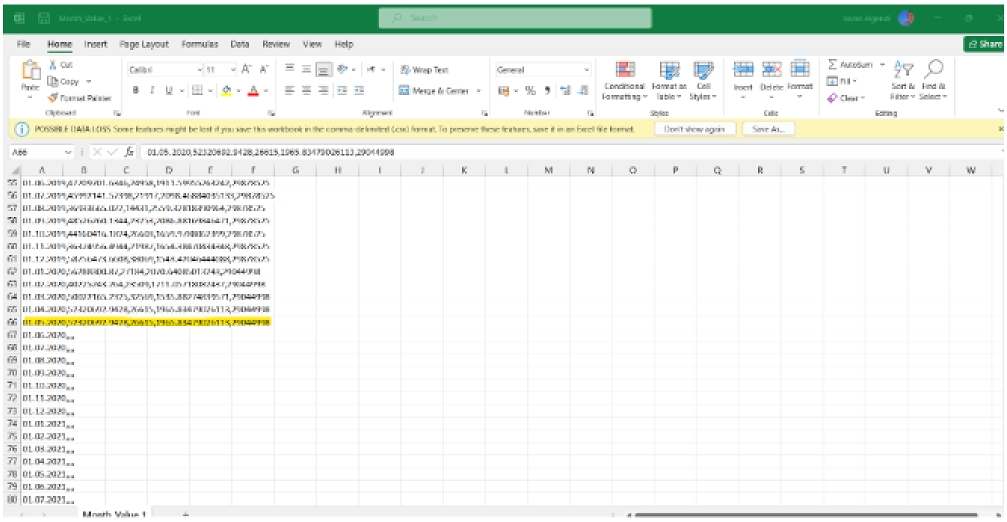


Notifications

Dismiss all

Generating ARM templates
Succeeded
a few seconds ago

Publishing completed
Successfully published from collaboration branch
a few seconds ago

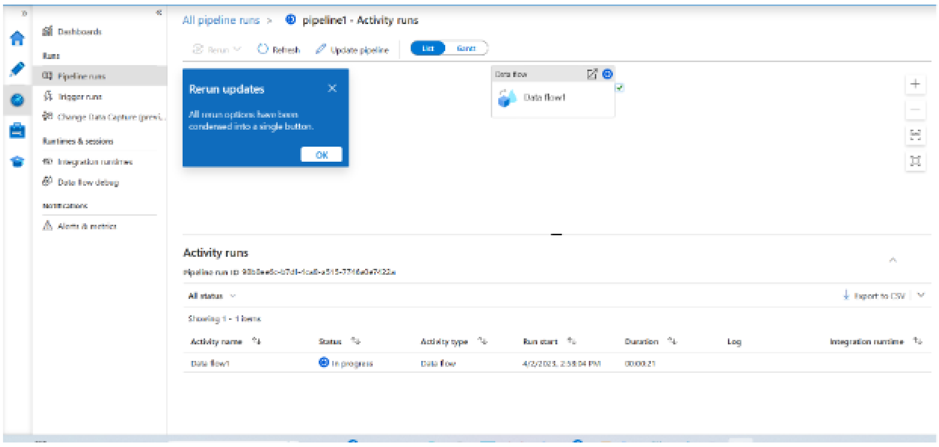


POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

A66 fx 01.05.2020,52320692.9428,26615,1965.83479026113,29044998

```
55 01.06.2019,[illegible]
56 01.07.2019,[illegible]
57 01.08.2019,[illegible]
58 01.09.2019,[illegible]
59 01.10.2019,[illegible]
60 01.11.2019,[illegible]
61 01.12.2019,[illegible]
62 01.01.2020,[illegible],29044998
63 01.02.2020,[illegible],29044998
64 01.03.2020,[illegible],29044998
65 01.04.2020,52320692.9428,26615,1965.83479026113,29044998
66 01.05.2020,52320692.9428,26615,1965.83479026113,29044998
67 01.06.2020,,,,
68 01.07.2020,,,,
69 01.08.2020,,,,
70 01.09.2020,,,,
71 01.10.2020,,,,
72 01.11.2020,,,,
73 01.12.2020,,,,
74 01.01.2021,,,,
75 01.02.2021,,,,
76 01.03.2021,,,,
77 01.04.2021,,,,
78 01.05.2021,,,,
79 01.06.2021,,,,
80 01.07.2021,,,,
```

Updating the original dataset by one row



All pipeline runs > pipeline1 - Activity runs

Rerun updates

All rerun options have been condensed into a single button.

OK

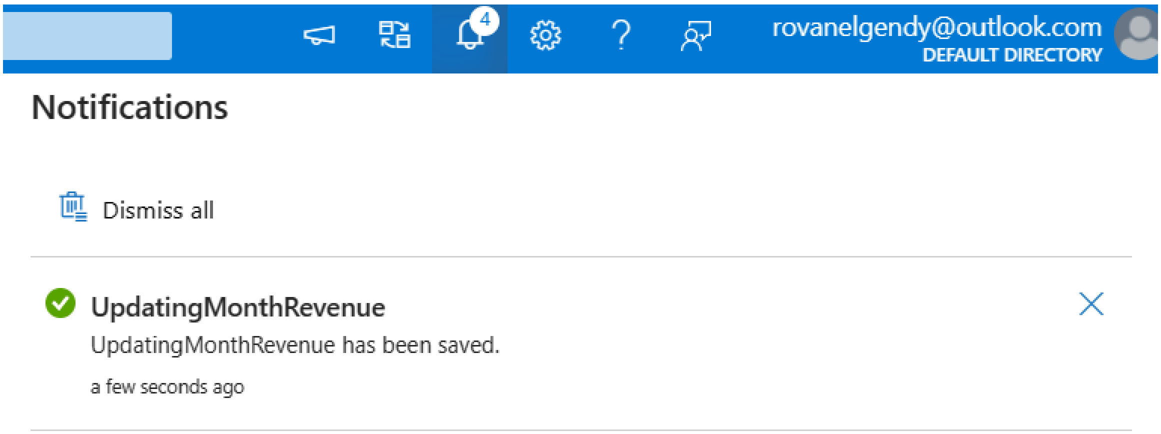
Activity runs

Pipeline run ID 93b0ee6c-b7d1-4ca8-a515-7746a0e7422e

Showing 1 - 1 items

Activity name	Status	Activity type	Run start	Duration	Log	Integration runtime
Data flow1	In progress	Data flow	4/2/2021, 2:58:04 PM	00:00:21		

pushing the trigger we made earlier manually



Notifications

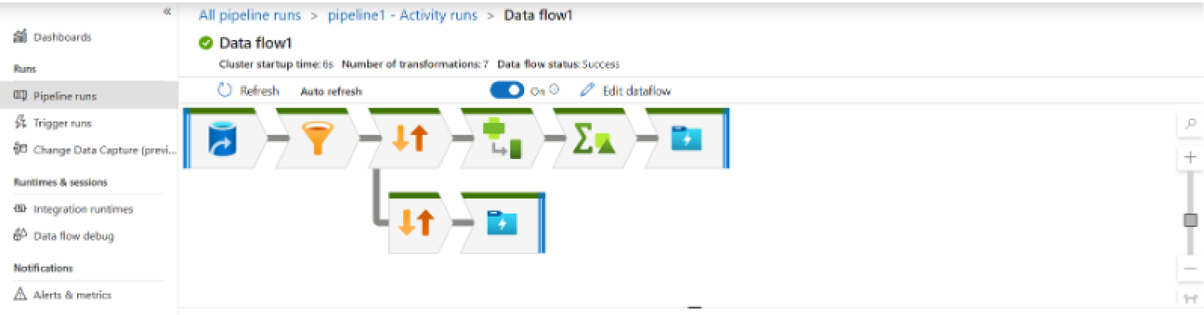
Dismiss all

UpdatingMonthRevenue

UpdatingMonthRevenue has been saved.

a few seconds ago

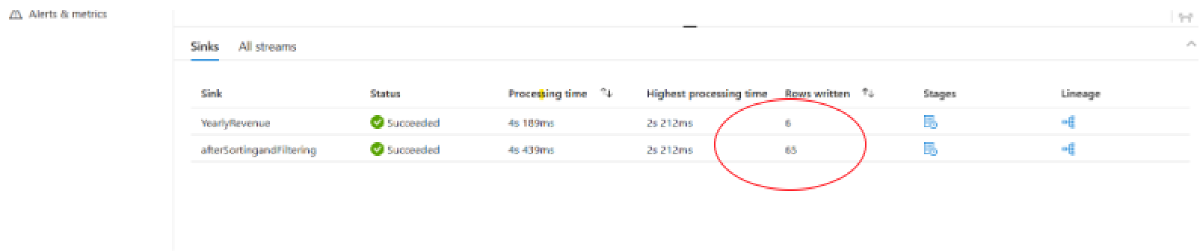
Trigger success notification



All pipeline runs > pipeline1 - Activity runs > Data flow1

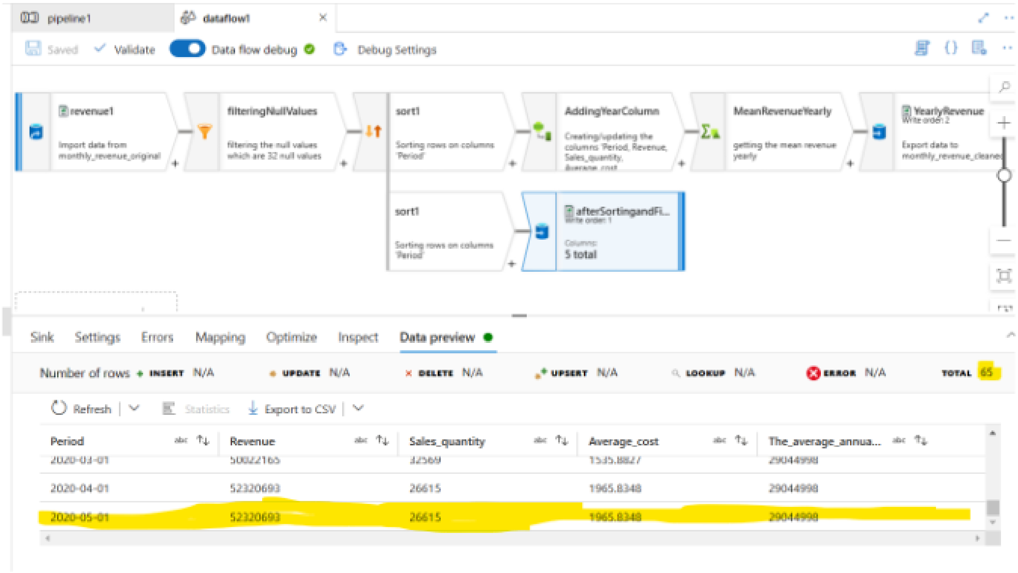
Data flow1

Cluster startup time: 6s Number of transformations: 7 Data flow status: Success



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		

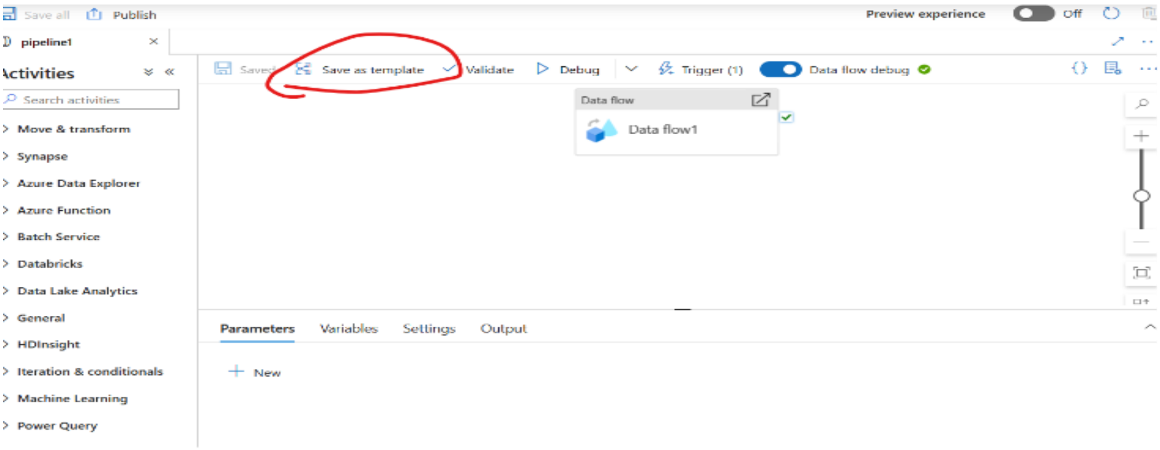
here we can see after trigger the rows read is updated to 65 rows instead of 64 rows


pipeline1
dataflow1
Saved
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'
afterSortingandFi...
Write order: 1
Columns:
5 total
Sink
Settings
Errors
Mapping
Optimize
Inspect
Data preview
Refresh
Statistics
Export to CSV

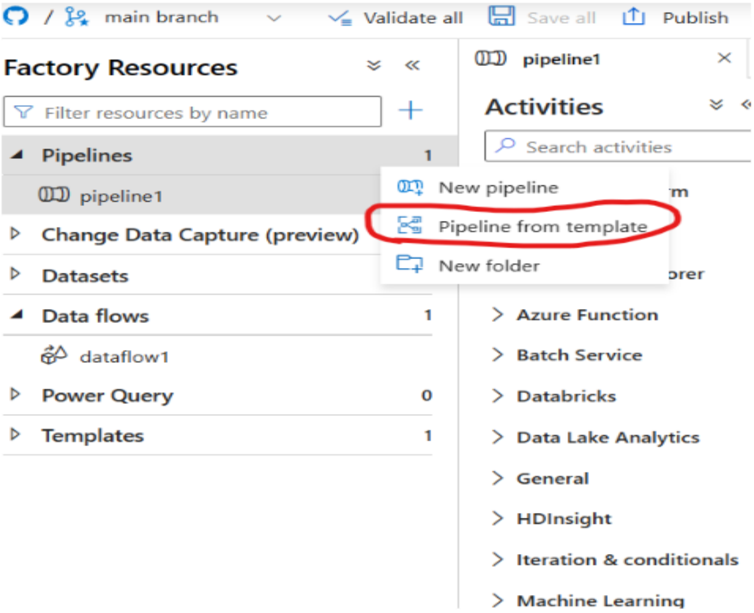
Number of rows INSERT N/A UPDATE N/A DELETE N/A UPSERT N/A LOOKUP N/A ERROR N/A TOTAL 65

Period	Revenue	Sales_quantity	Average_cost	The_average_annua...
2020-03-01	50022165	32569	1535.8827	29044998
2020-04-01	52320693	26615	1965.8348	29044998
2020-05-01	52320693	26615	1965.8348	29044998

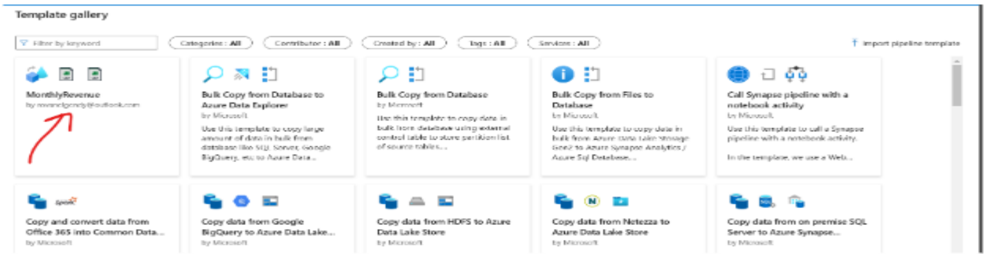
the updated row in the data preview



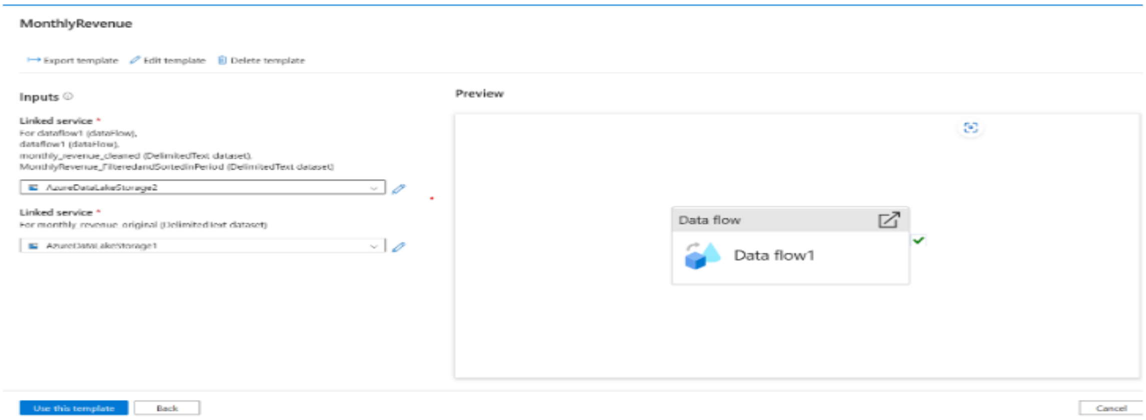
You can save the pipeline as template to import later



importing the tempelate


Template gallery
Filter by keyword
Categories : All
Contributor : All
Created by : All
Tags : All
Services : All
Import pipeline template
MonthlyRevenue
Bulk Copy from Database to Azure Data Explorer
by Microsoft
Bulk Copy from Database
by Microsoft
Bulk Copy from Files to Database
by Microsoft
Call Synapse pipeline with a notebook activity
by Microsoft
Copy and convert data from Office 365 into Common Data...
by Microsoft
Copy data from Google BigQuery to Azure Data Lake...
by Microsoft
Copy data from HDFS to Azure Data Lake Store
by Microsoft
Copy data from Netezza to Azure Data Lake Store
by Microsoft
Copy data from on premise SQL Server to Azure Synapse...
by Microsoft

the tempelate gallery


MonthlyRevenue

Inputs

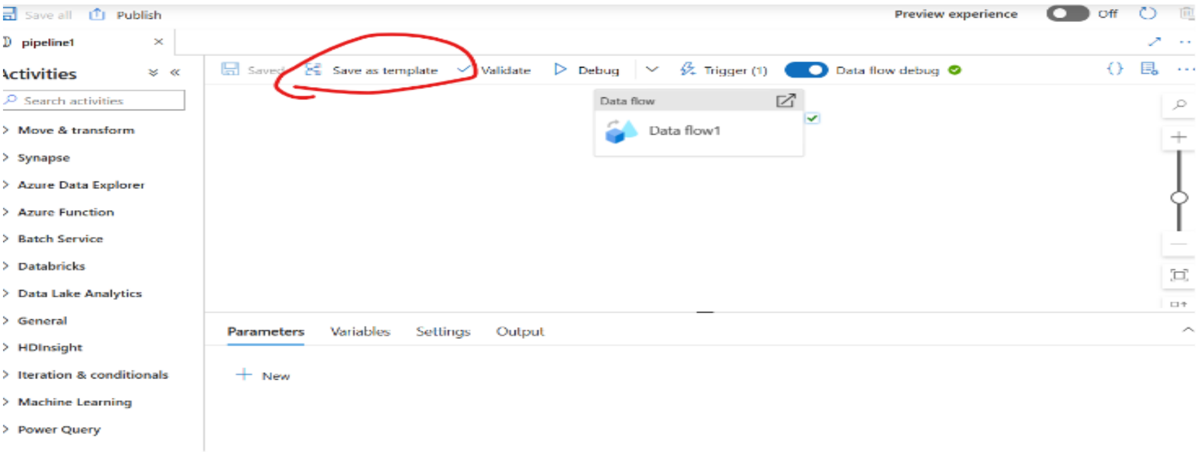
Linked service *
For dataflow1 (dataFlow),
dataflow1 (dataFlow),
monthly_revenue_cleaned (DelimitedText dataset),
MonthlyRevenue_FilteredandSortedinPeriod (DelimitedText dataset)

AzureDataLakeStorage2

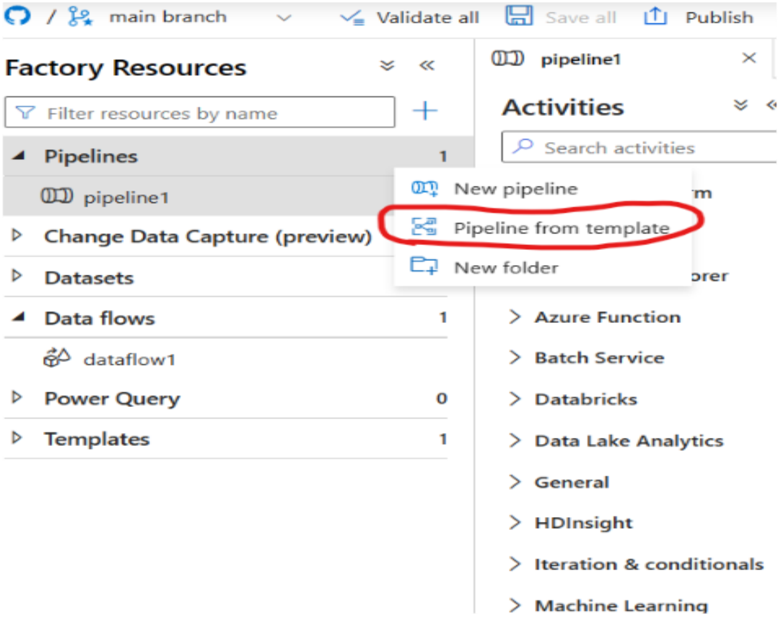
Linked service *
For monthly_revenue_original (DelimitedText dataset)

AzureDataLakeStorage1

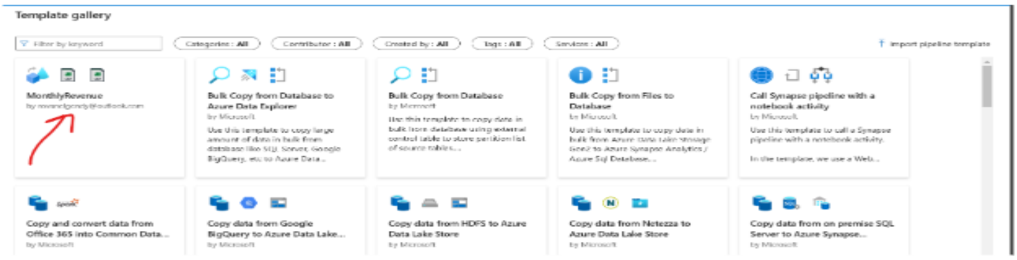
last step to click on use this tempelate



You can save the pipeline as template to import later



importing the tempelate


Template gallery
Filter by keyword
Categories : All
Contributor : All
Created by : All
Tags : All
Services : All
Import pipeline template
MonthlyRevenue
Bulk Copy from Database to Azure Data Explorer
by Microsoft
Bulk Copy from Database
by Microsoft
Bulk Copy from Files to Database
by Microsoft
Call Synapse pipeline with a notebook activity
by Microsoft
Copy and convert data from Office 365 into Common Data...
by Microsoft
Copy data from Google BigQuery to Azure Data Lake...
by Microsoft
Copy data from HDFS to Azure Data Lake Store
by Microsoft
Copy data from Netezza to Azure Data Lake Store
by Microsoft
Copy data from on premise SQL Server to Azure Synapse...
by Microsoft

Use this template to copy large amount of data in bulk from database like SQL Server, Google BigQuery, etc to Azure Data...

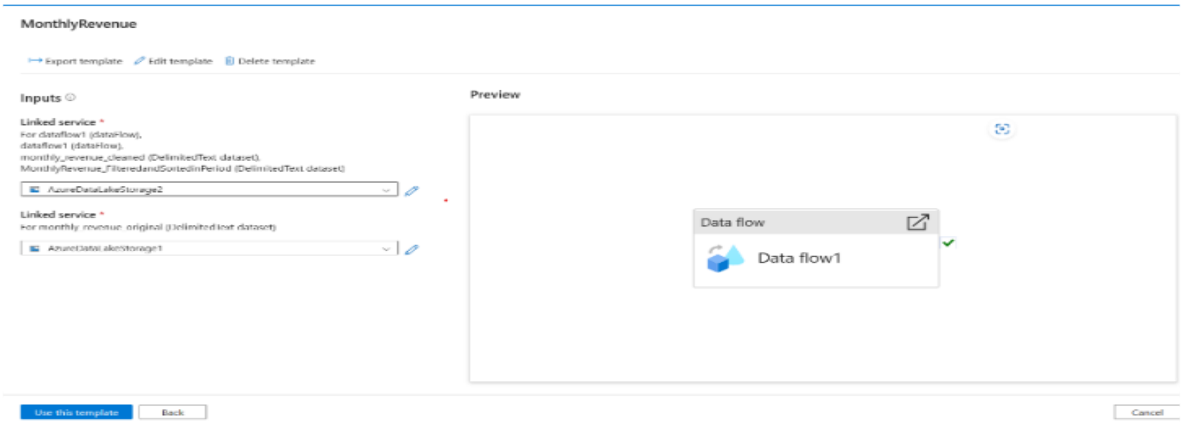
Use this template to copy data in bulk from database using external control table to store partition list of source tables...

Use this template to copy data in bulk from Azure Data Lake Storage Gen2 to Azure Synapse Analytics / Azure Sql Database...

Use this template to call a Synapse pipeline with a notebook activity.

In the template, we use a Web...

the tempelate gallery



Inputs

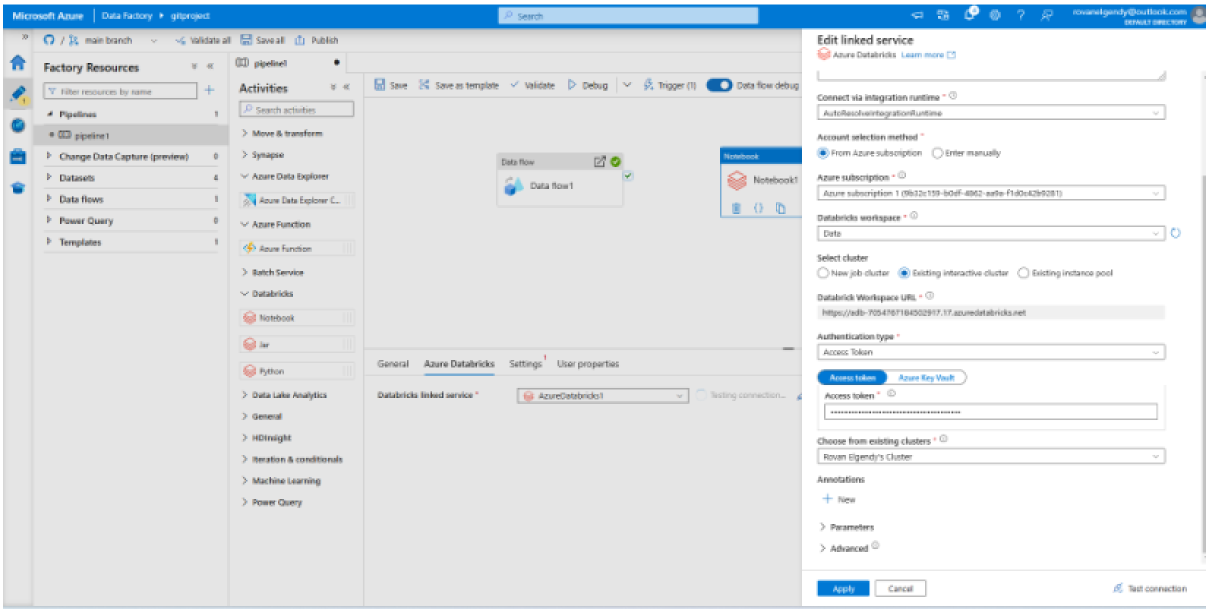
Linked service *
For dataflow1 (dataFlow),
dataflow1 (dataFlow),
monthly_revenue_cleaned (DelimitedText dataset),
MonthlyRevenue_FilteredandSortedinPeriod (DelimitedText dataset)

AzureDataLakeStorage2

Linked service *
For monthly_revenue_original (DelimitedText dataset)

AzureDataLakeStorage1

last step to click on use this tempelate



Edit linked service

Azure Databricks

Connect via integration runtime *: AutoResolveIntegrationRuntime

Account selection method *: ● From Azure subscription ○ Enter manually

Azure subscription *: Azure subscription 1 (9b32c159-b0df-4862-aa9a-f1d0c42b9281)

Databricks workspace *: Data

Select cluster: ○ New job cluster ● Existing interactive cluster ○ Existing instance pool

Databrick Workspace URL *: https://adb-7054787184502917.17.azuredatabricks.net

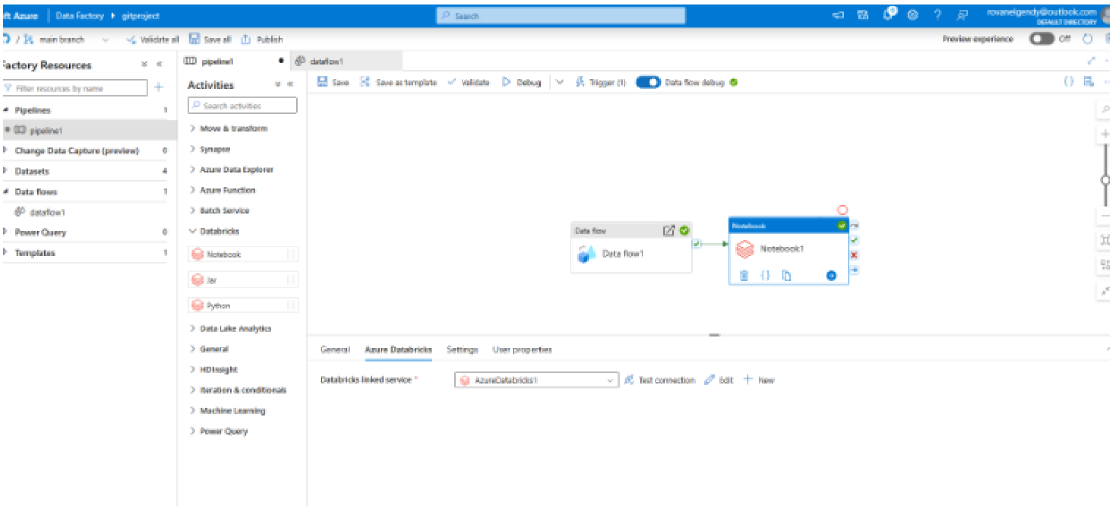
Authentication type *: Access Token

Access token *: ••••••••••••••••••••••••••••••••••••

Choose from existing clusters *: Rovan Elgendy's Cluster

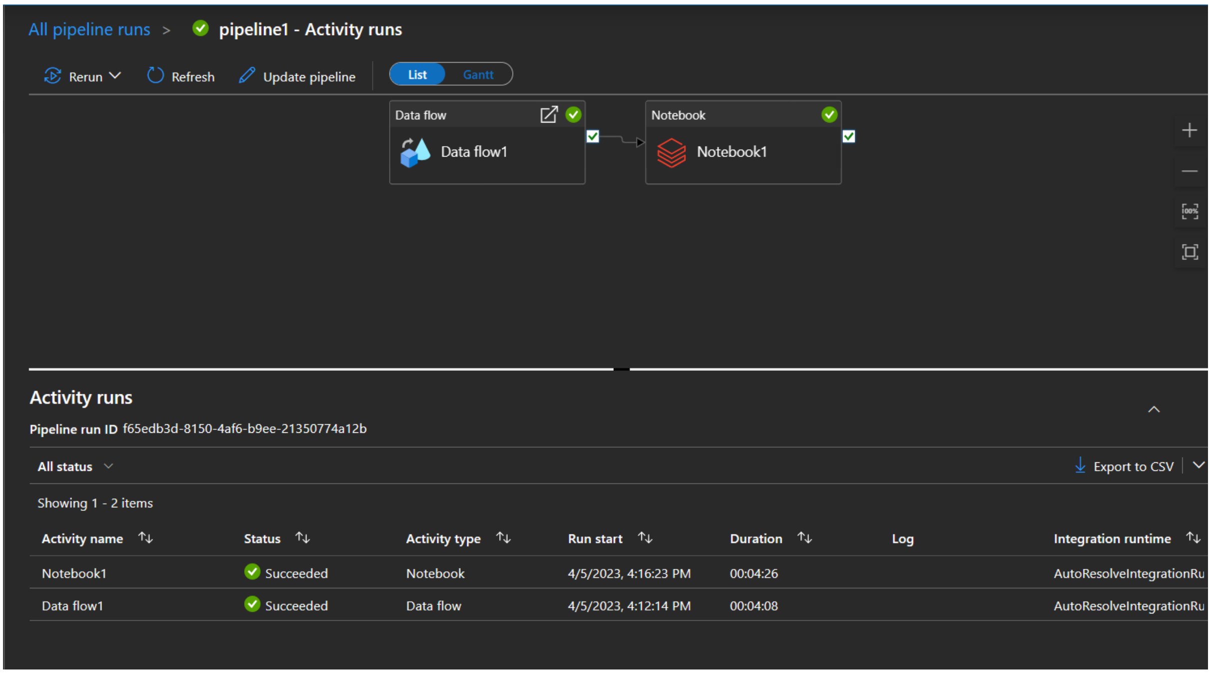
Databricks linked service *: AzureDatabricks1

Adding Databricks Service using existing cluster and the access token



Databricks linked service *: AzureDatabricks1

Adding databricks notebook -which is already made in different file- to our dataflow



Activity runs

Pipeline run ID f65edb3d-8150-4af6-b9ee-21350774a12b

All status

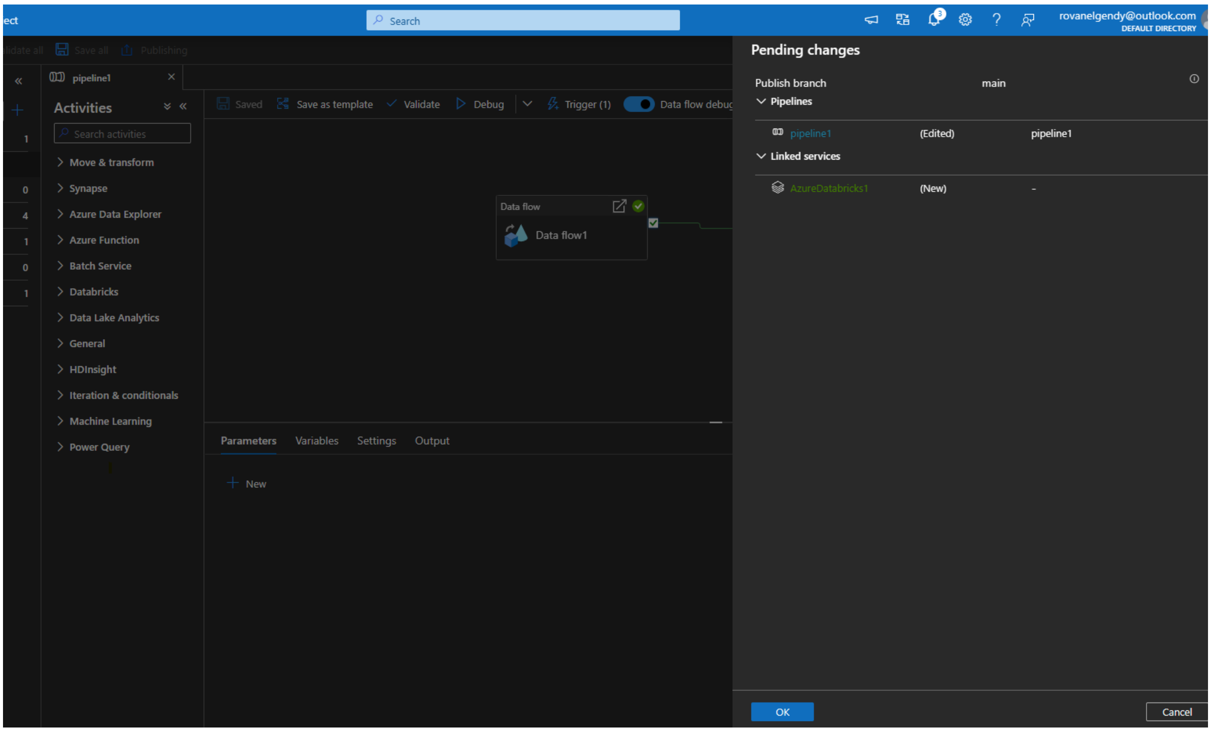
Export to CSV

Showing 1 - 2 items

Activity name	Status	Activity type	Run start	Duration	Log	Integration runtime
Notebook1	✓ Succeeded	Notebook	4/5/2023, 4:16:23 PM	00:04:26		AutoResolveIntegrationRu
Data flow1	✓ Succeeded	Data flow	4/5/2023, 4:12:14 PM	00:04:08		AutoResolveIntegrationRu

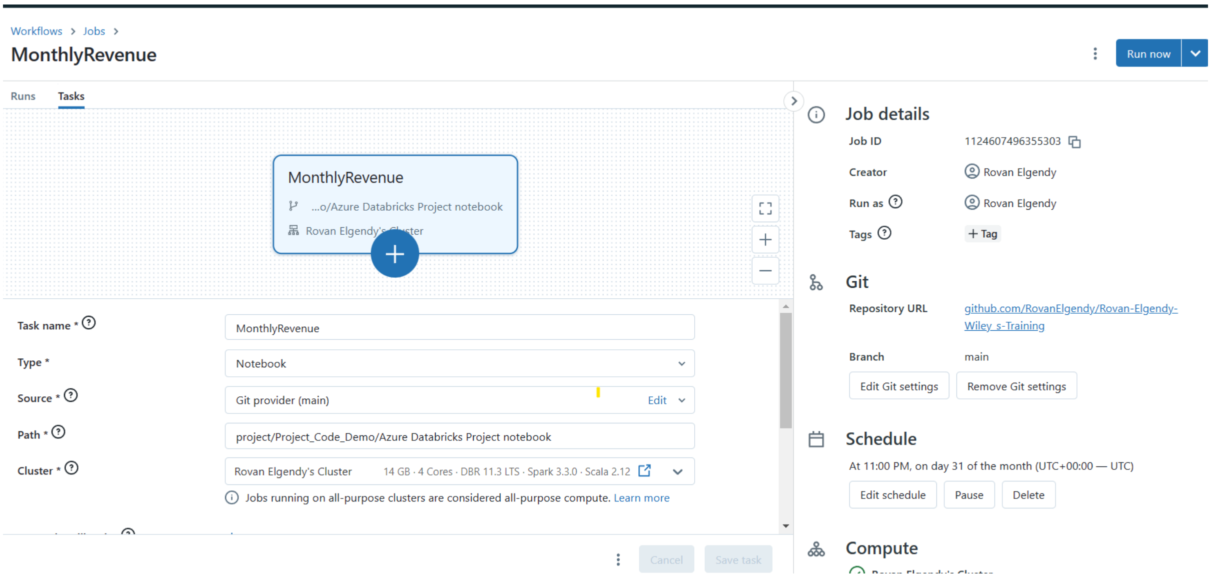
Triggering the pipeline after updating the original dataset with one more row:

- The dataflow is updated
- The notebook is updated as well.



Pending changes

Publish branch	main	
Pipelines		
pipeline1	(Edited)	pipeline1
Linked services		
AzureDatabricks1	(New)	-



MonthlyRevenue

Task name *	MonthlyRevenue
Type *	Notebook
Source *	Git provider (main) Edit
Path *	project/Project_Code_Demo/Azure Databricks Project notebook
Cluster *	Rovan Elgendy's Cluster 14 GB · 4 Cores · DBR 11.3 LTS · Spark 3.3.0 · Scala 2.12

Jobs running on all-purpose clusters are considered all-purpose compute. Learn more

Job details

Job ID	1124607496355303
Creator	Rovan Elgendy
Run as	Rovan Elgendy
Tags	+ Tag

Git

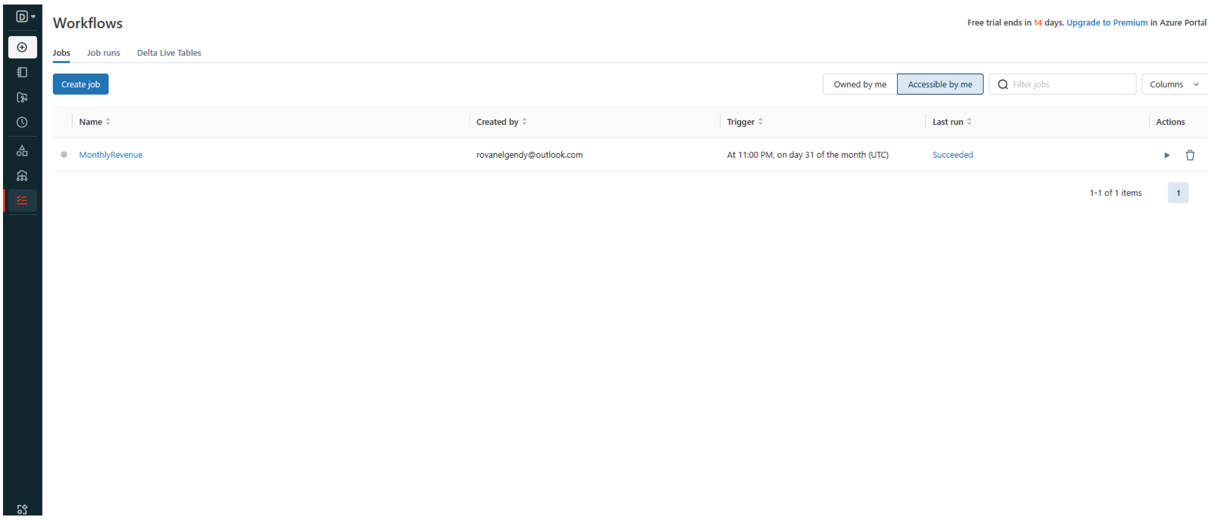
Repository URL	github.com/RovanElgendy/Rovan-Elgendy-Wiley_s-Training
Branch	main

Schedule

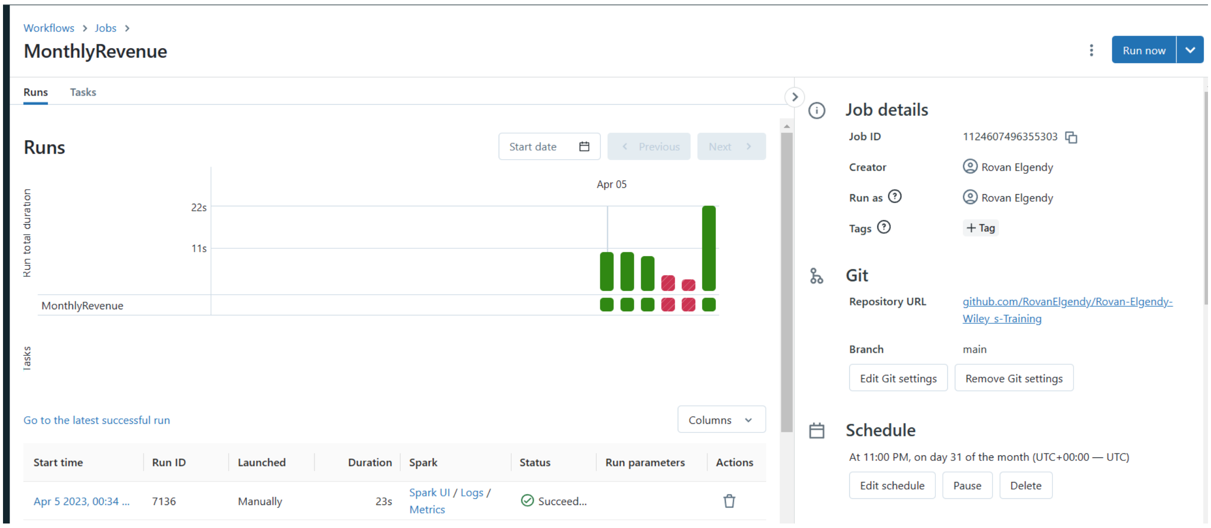
At 11:00 PM, on day 31 of the month (UTC+00:00 — UTC)

Compute

creating workflow job in databricks with Git Repo integration to the databricks notebook located in the Repo. then trigger it.



Name	Created by	Trigger	Last run	Actions
MonthlyRevenue	rovanelgendy@outlook.com	At 11:00 PM, on day 31 of the month (UTC)	Succeeded	



Start time	Run ID	Launched	Duration	Spark	Status	Run parameters	Actions
Apr 5 2023, 00:34 ...	7136	Manually	23s	Spark UI / Logs / Metrics	Succeed...		

Job details

Job ID 1124607496355303

Creator Rovan Elgendy

Run as Rovan Elgendy

Tags + Tag

Git

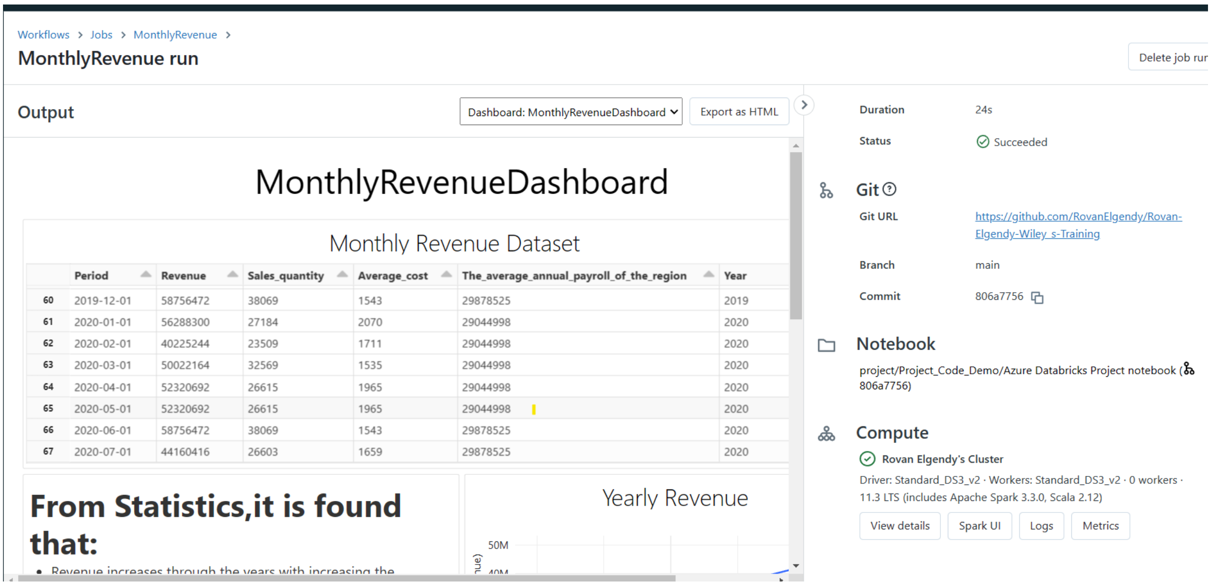
Repository URL github.com/RovanElgendy/Rovan-Elgendy-Wiley_s-Training

Branch main

Schedule

At 11:00 PM, on day 31 of the month (UTC+00:00 — UTC)

It shows the updated workflow of the notebook and gets updated in Git Repo.



Workflows > Jobs > MonthlyRevenue >

MonthlyRevenue run

Output

MonthlyRevenueDashboard

Monthly Revenue Dataset

	Period	Revenue	Sales_quantity	Average_cost	The_average_annual_payroll_of_the_region	Year
60	2019-12-01	58756472	38069	1543	29878525	2019
61	2020-01-01	56288300	27184	2070	29044998	2020
62	2020-02-01	40225244	23509	1711	29044998	2020
63	2020-03-01	50022164	32569	1535	29044998	2020
64	2020-04-01	52320692	26615	1965	29044998	2020
65	2020-05-01	52320692	26615	1965	29044998	2020
66	2020-06-01	58756472	38069	1543	29878525	2020
67	2020-07-01	44160416	26603	1659	29878525	2020

From Statistics,it is found that:

- Revenue increases through the years with increasing the

Yearly Revenue

Duration	24s
Status	✓ Succeeded

Git

Git URL	https://github.com/RovanElgendy/Rovan-Elgendy-Wiley_s-Training
Branch	main
Commit	806a7756

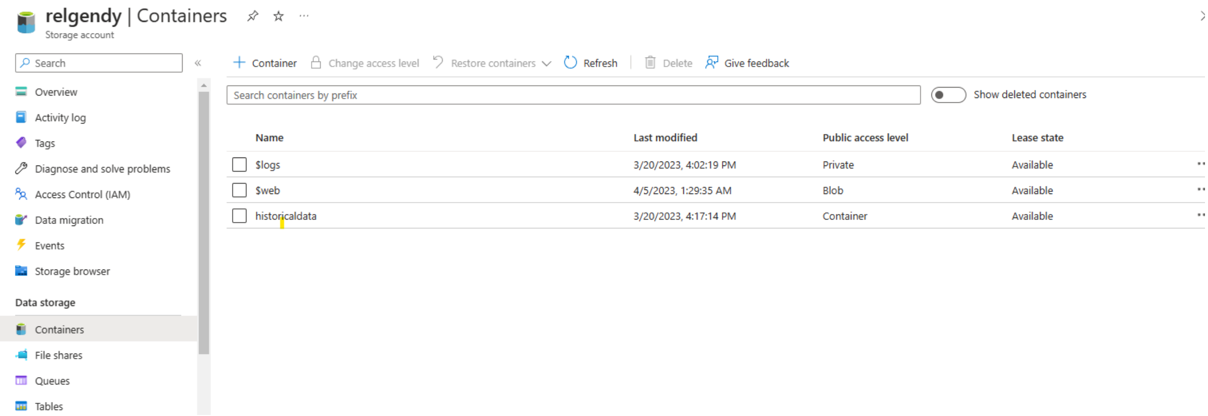
Notebook

project/Project_Code_Demo/Azure Databricks Project notebook (806a7756)

Compute

✓ Rovan Elgendy's Cluster

Driver: Standard_DS3_v2 · Workers: Standard_DS3_v2 · 0 workers · 11.3 LTS (includes Apache Spark 3.3.0, Scala 2.12)



relgendy | Containers

Storage account

Name	Last modified	Public access level	Lease state
$logs	3/20/2023, 4:02:19 PM	Private	Available
$web	4/5/2023, 1:29:35 AM	Blob	Available
historicaldata	3/20/2023, 4:17:14 PM	Container	Available

creating $web container in blob storage to store web files .html to present the workflow of the project.