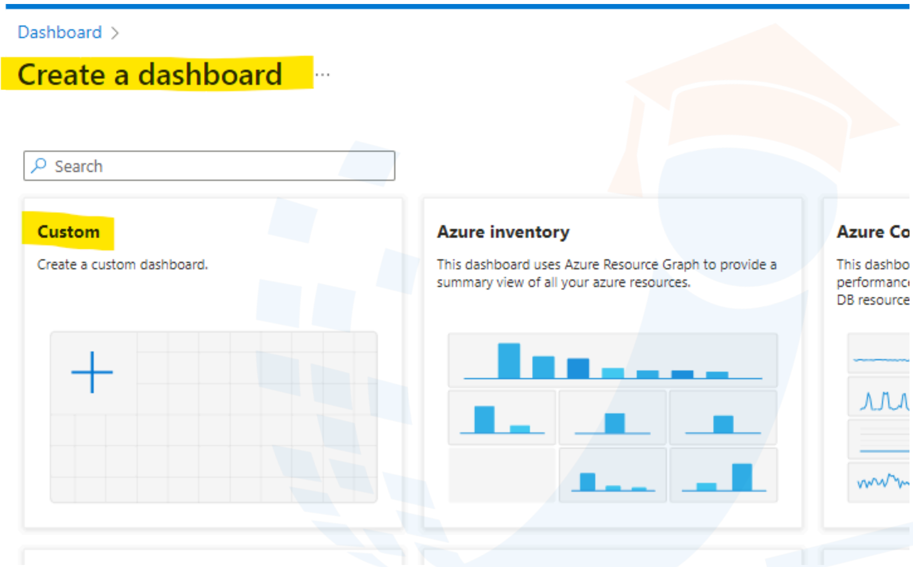
Use-case 2 : Utilize Azure Data Factory (ADF) to ingest Orders and Customers data, and execute fundamental transformations on the datasets.

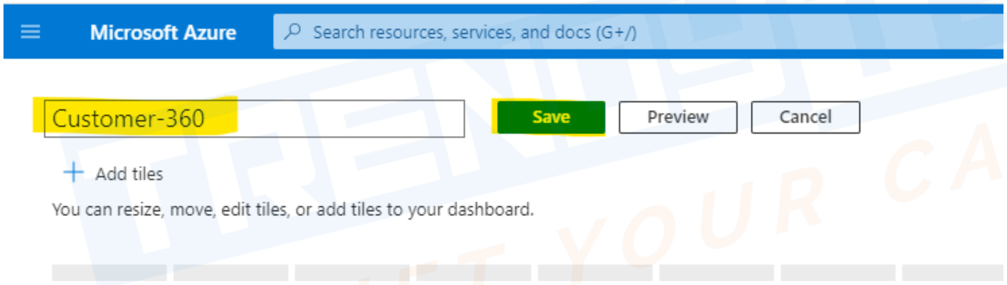
Note: You can create a Dashboard for the project to organize the resources related to the project at one place.

Click on Dashboard => Create => Custom =>



Create a custom dashboard.

This dashboard uses Azure Resource Graph to provide a summary view of all your azure resources.



Customer-360

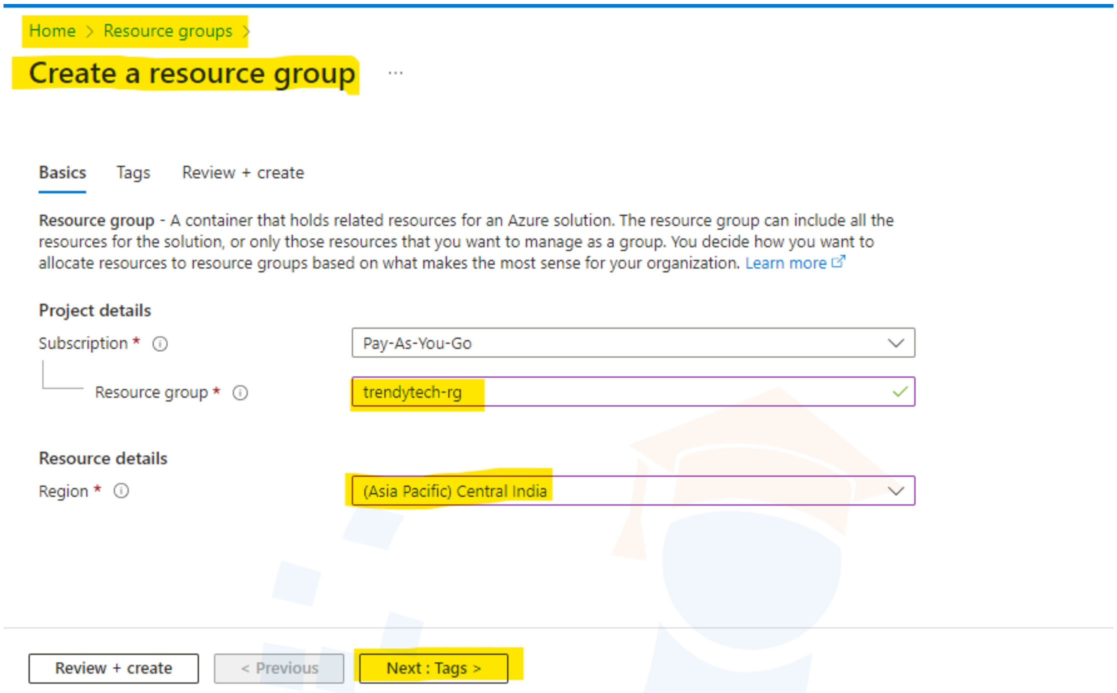
You can resize, move, edit tiles, or add tiles to your dashboard.

Dataset orders.csv can be downloaded from - https://files.cdn.thinkific.com/file_uploads/349536/attachments/c28/5fb/25b/orders.csv

Dataset customers.csv is provided in the Week18 downloadable section in the portal

Requirement - Ingest orders.csv file from external URL to ADLS Gen2

Create Resource Group project Resources:



Create a resource group

Resource group - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. Learn more

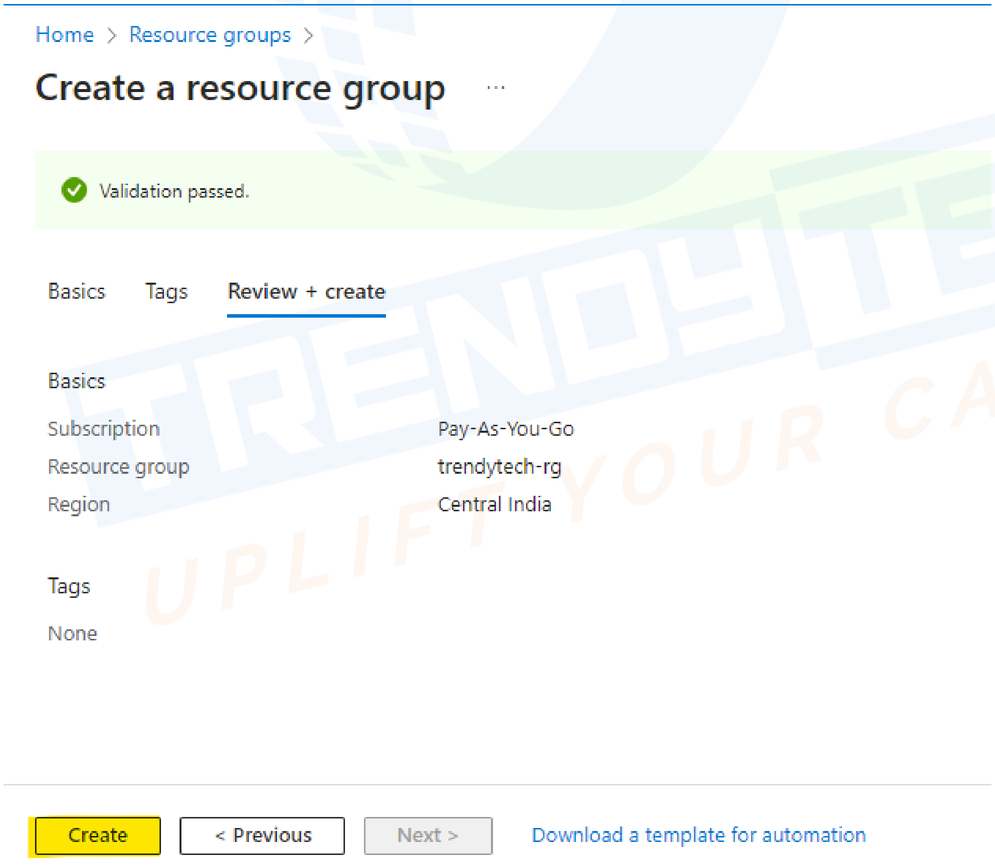
Project details

Subscription * : Pay-As-You-Go

Resource group * : trendytech-rg

Resource details

Region * : (Asia Pacific) Central India



Create a resource group

Validation passed.

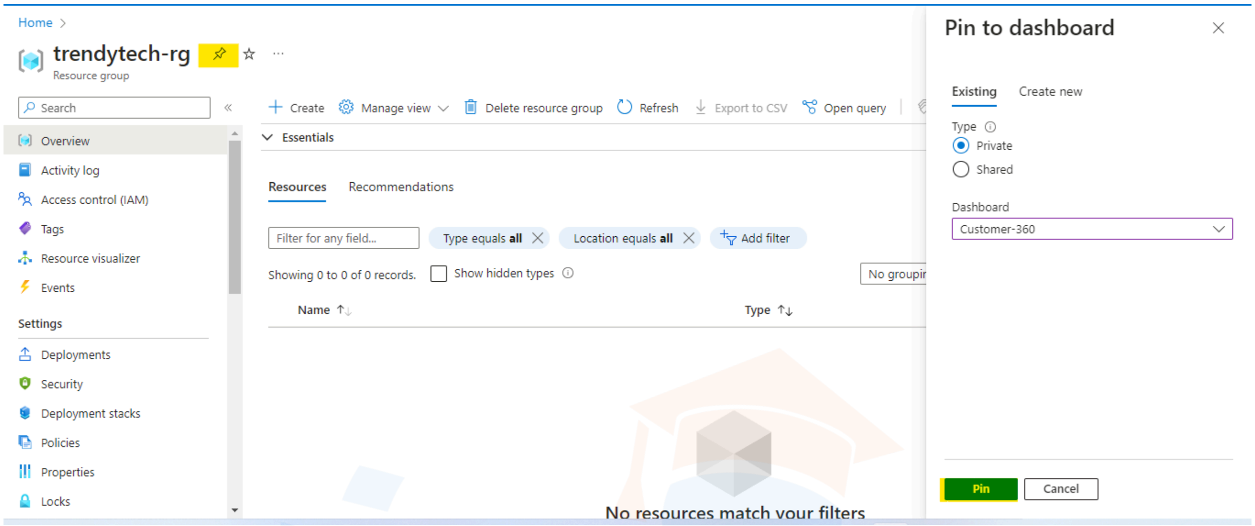
Basics

Subscription	Pay-As-You-Go
Resource group	trendytech-rg
Region	Central India

Tags

None

Now you can pin it to a dashboard as shown below.

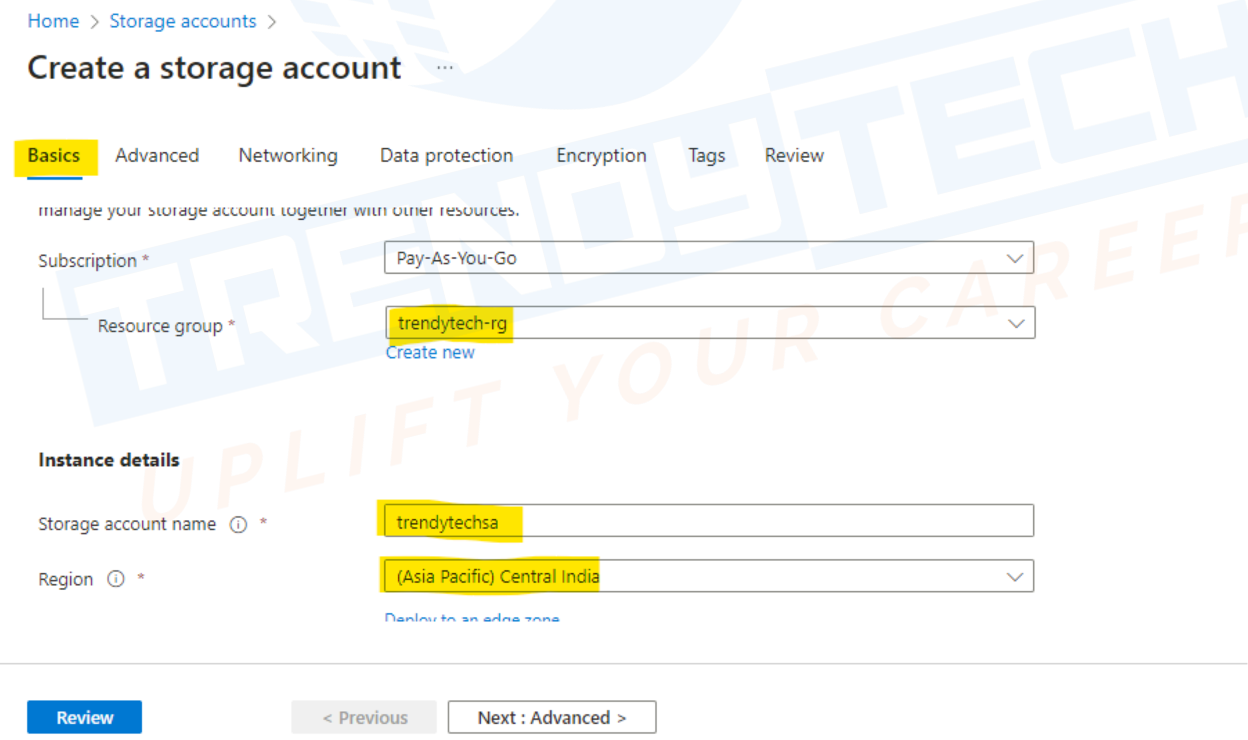


Type: ● Private ○ Shared

Dashboard: Customer-360

Create a Resource - Storage Account (Datalake) :

Create a Storage Account (Enable Hierarchical namespace to make it as data lake storage and not just the blob storage)



manage your storage account together with other resources.

Subscription * Pay-As-You-Go

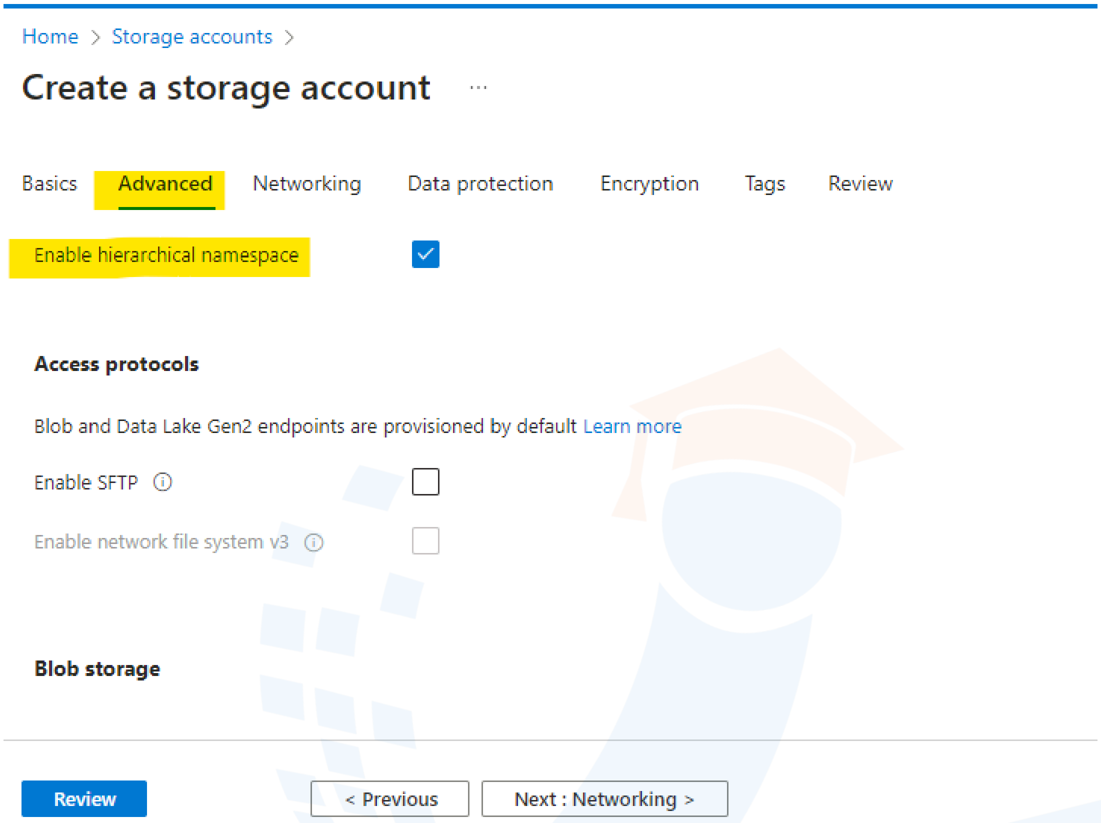
Resource group * trendytech-rg

Instance details

Storage account name * trendytechsa

Region * (Asia Pacific) Central India

Keep other option as it is just **"Enable hierarchical namespace"** refer attached screenshot.



Enable hierarchical namespace ☑

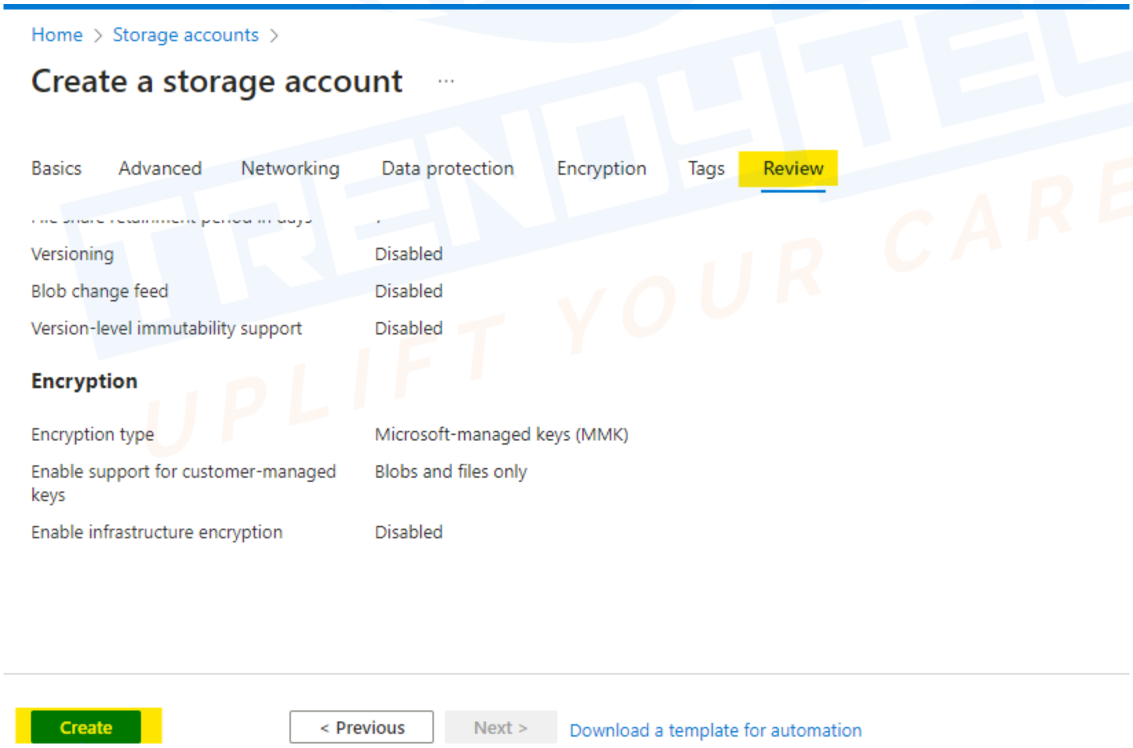
Access protocols

Blob and Data Lake Gen2 endpoints are provisioned by default Learn more

Enable SFTP ⓘ □

Enable network file system v3 ⓘ □

Blob storage



Home > Storage accounts >
Create a storage account ...
Basics
Advanced
Networking
Data protection
Encryption
Tags
Review
Create
< Previous
Next >
Download a template for automation

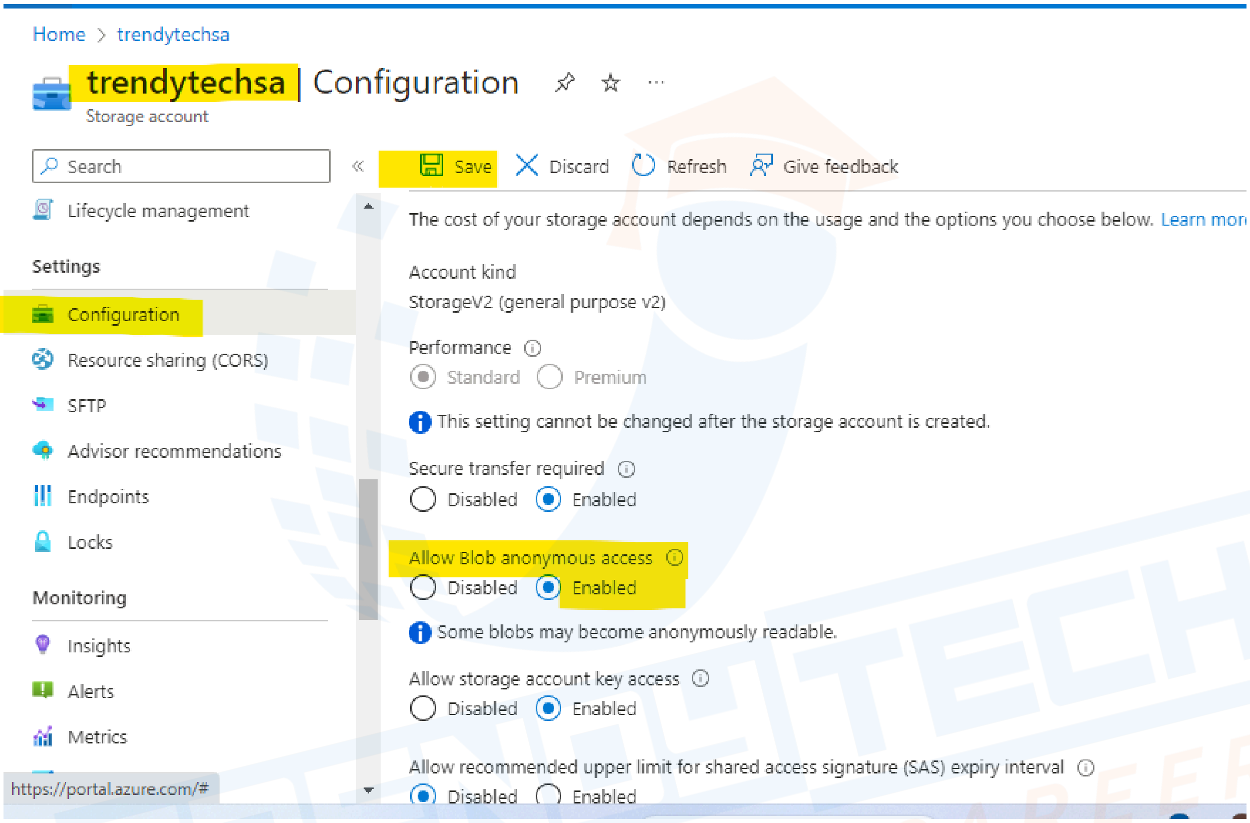
Versioning	Disabled
Blob change feed	Disabled
Version-level immutability support	Disabled

Encryption

Encryption type	Microsoft-managed keys (MMK)
Enable support for customer-managed keys	Blobs and files only
Enable infrastructure encryption	Disabled

Note: Once a resource has been created, you have the option to add it to the dashboard for quick access.

Note : After creating resources go to Setting => Configuration => Enable (Allow Blob anonymous access) => click on “Save”



The cost of your storage account depends on the usage and the options you choose below. Learn more

Account kind
StorageV2 (general purpose v2)

Performance ⓘ
● Standard ○ Premium

This setting cannot be changed after the storage account is created.

Secure transfer required ⓘ
○ Disabled ● Enabled

Allow Blob anonymous access ⓘ
○ Disabled ● Enabled

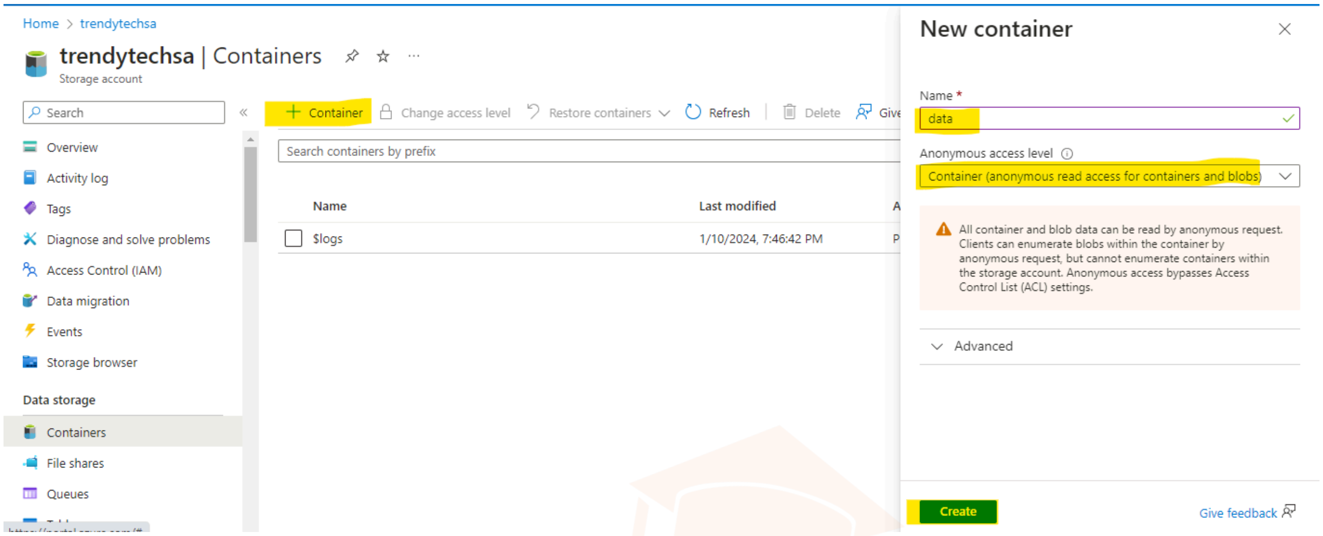
Some blobs may become anonymously readable.

Allow storage account key access ⓘ
○ Disabled ● Enabled

Allow recommended upper limit for shared access signature (SAS) expiry interval ⓘ
● Disabled ○ Enabled

Now create the container “data” and create the directory “output”

Note: Sumit explains the creation of the container and output folder later in the video while setting up the linked service; however, the steps are provided here for your convenience.



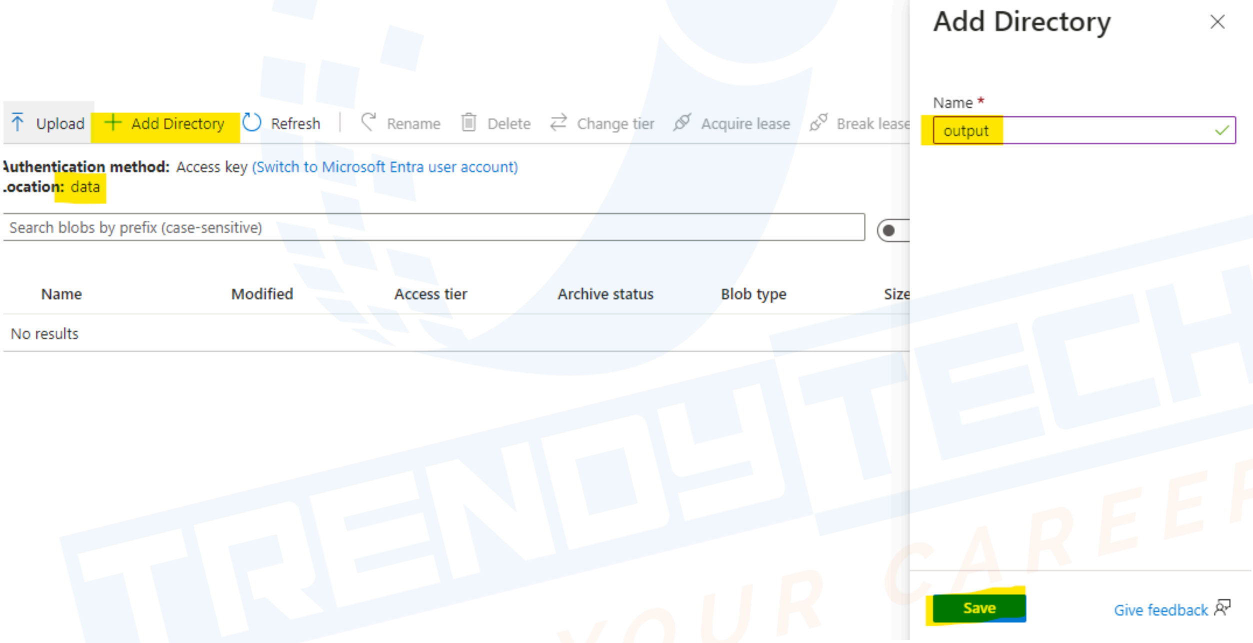
Name	Last modified	A
$logs	1/10/2024, 7:46:42 PM	P

New container

Name *: data

Anonymous access level: Container (anonymous read access for containers and blobs)

All container and blob data can be read by anonymous request. Clients can enumerate blobs within the container by anonymous request, but cannot enumerate containers within the storage account. Anonymous access bypasses Access Control List (ACL) settings.

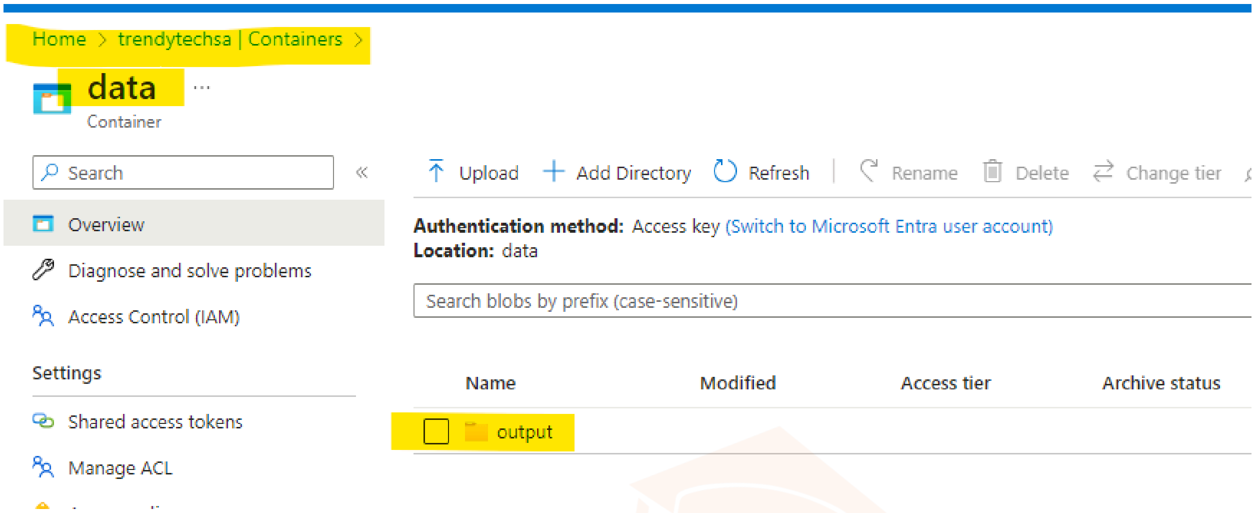


Authentication method: Access key (Switch to Microsoft Entra user account)
Location: data

Name	Modified	Access tier	Archive status	Blob type	Size
No results					

Add Directory

Name *: output



Home > trendytechsa | Containers >

data ...

Container

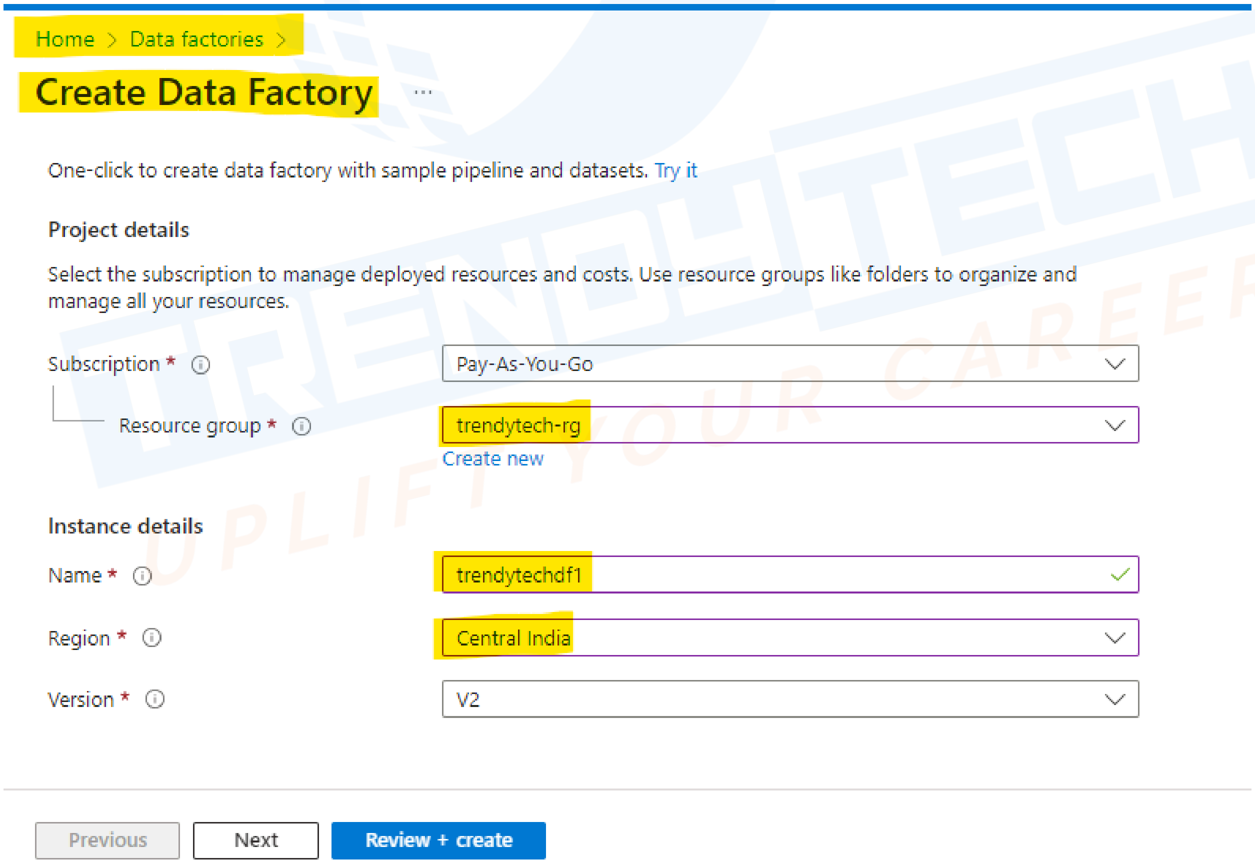
Authentication method: Access key (Switch to Microsoft Entra user account)
Location: data

Search blobs by prefix (case-sensitive)

Name	Modified	Access tier	Archive status
□ output			

Create a Resource - Azure Data Factory:

Please consult the attached screenshot for guidance on setting up Azure Data Factory.



Home > Data factories >

Create Data Factory ...

One-click to create data factory with sample pipeline and datasets. Try it

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : Pay-As-You-Go

Resource group * : trendytech-rg

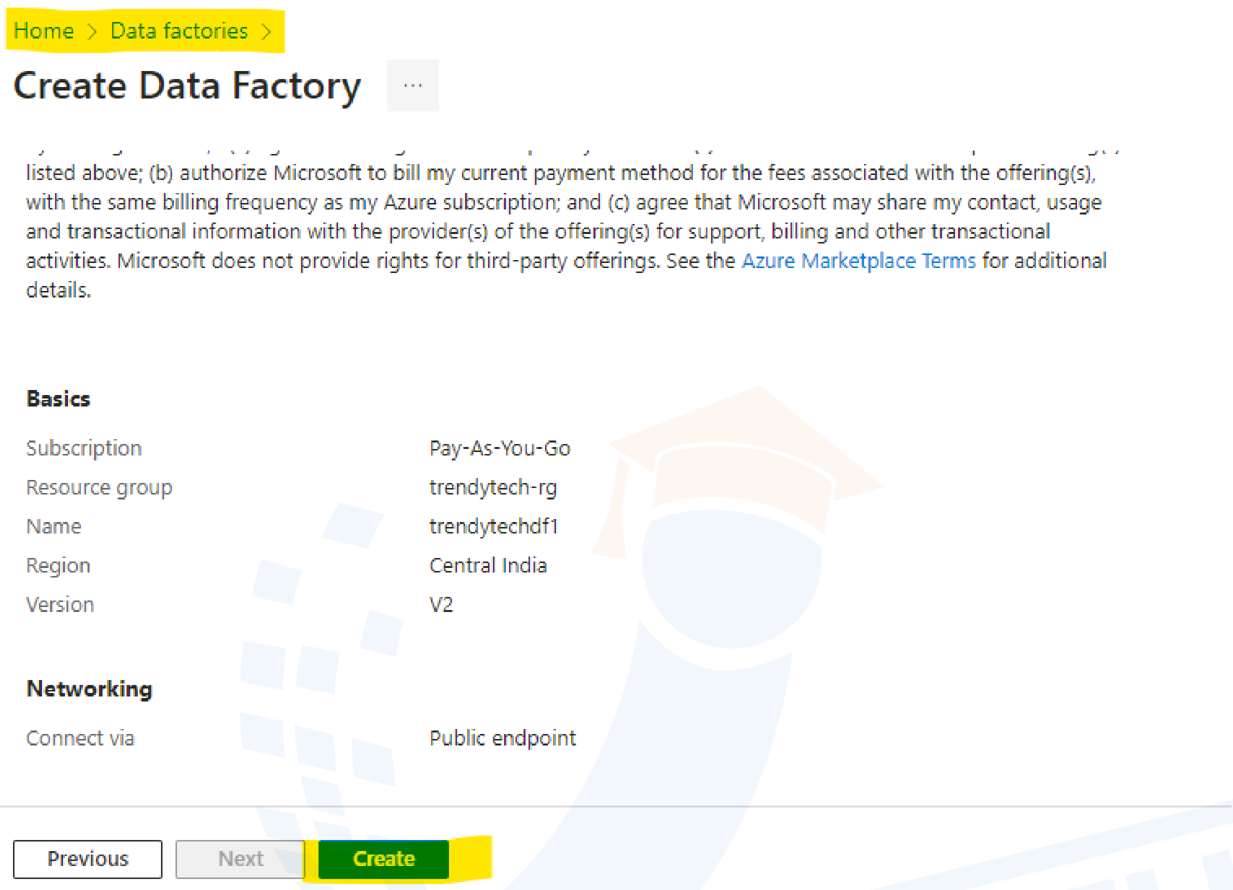
Create new

Instance details

Name * : trendytechdf1

Region * : Central India

Version * : V2



Home > Data factories >

Create Data Factory

listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the Azure Marketplace Terms for additional details.

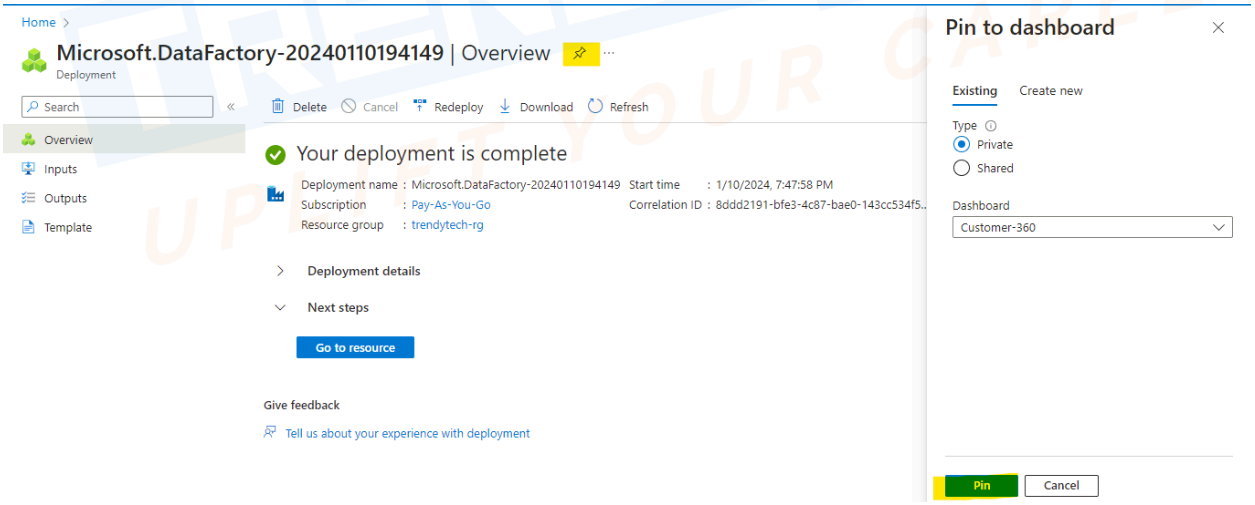
Basics

Subscription	Pay-As-You-Go
Resource group	trendytech-rg
Name	trendytechdf1
Region	Central India
Version	V2

Networking

Connect via	Public endpoint

Note: Again pin this resource to the same dashboard that you have created.



Home >

Microsoft.DataFactory-20240110194149 | Overview

Deployment

Your deployment is complete

Deployment name	: Microsoft.DataFactory-20240110194149	Start time	: 1/10/2024, 7:47:58 PM
Subscription	: Pay-As-You-Go	Correlation ID	: 8ddd2191-bfe3-4c87-bae0-143cc534f5...
Resource group	: trendytech-rg		

Give feedback

Tell us about your experience with deployment

Pin to dashboard

Type

● Private

○ Shared

Dashboard

Customer-360

Create a Linked Service for Source (choose HTTP connector):

orders.csv can be downloaded from -
https://files.cdn.thinkific.com/file_uploads/349536/attachments/c28/5fb/25b/orders.csv

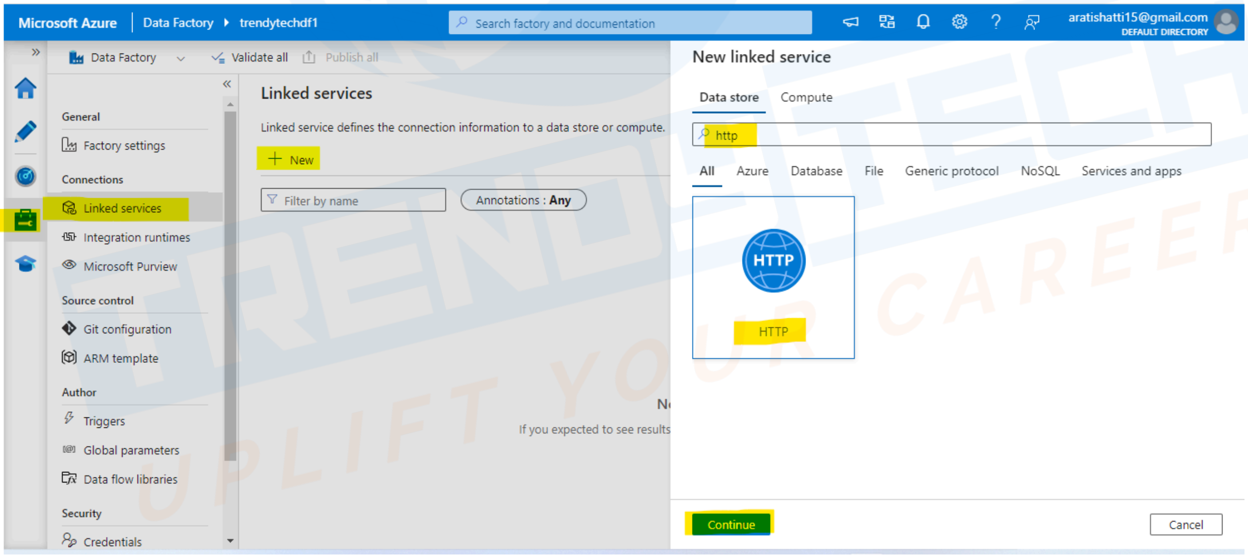
BaseURL - https://files.cdn.thinkific.com

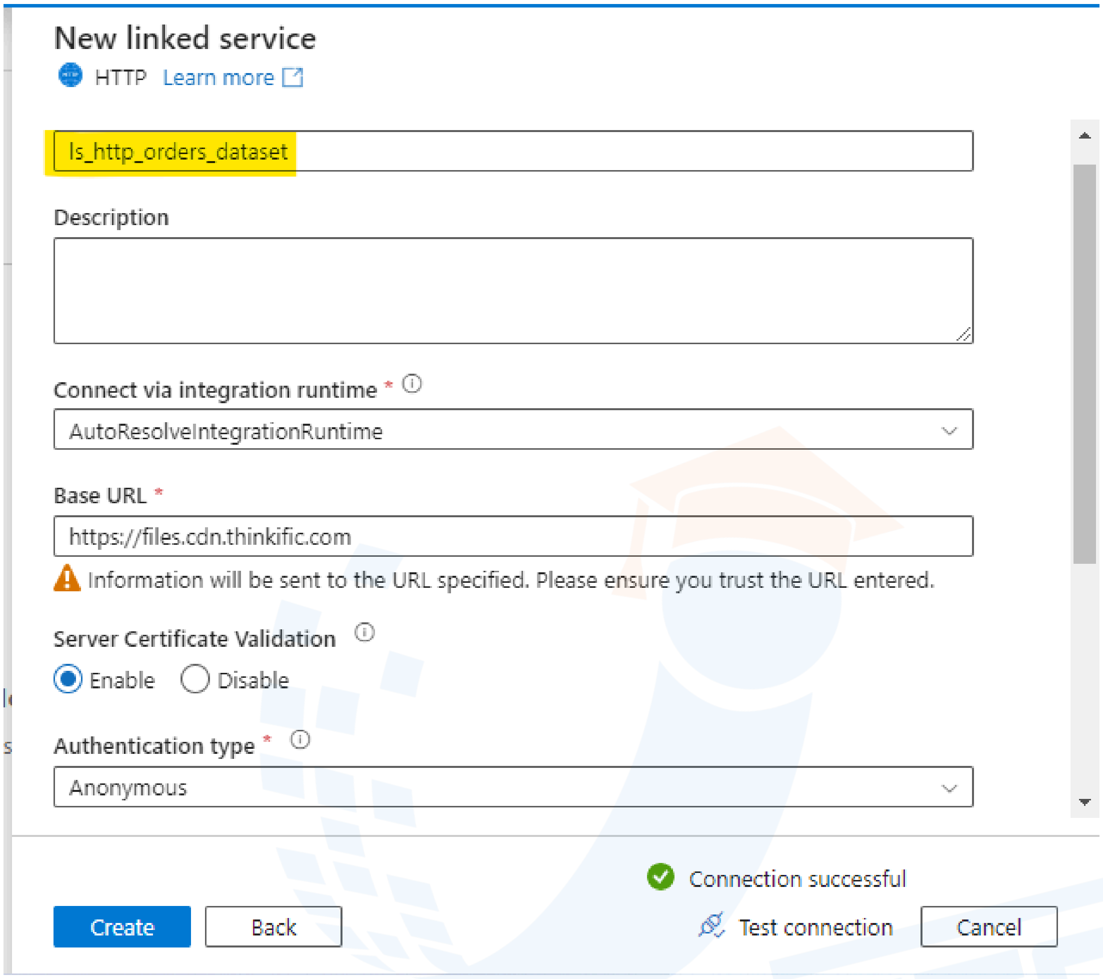
Relative URL - file_uploads/349536/attachments/c28/5fb/25b/orders.csv

Note: Launch the Azure Data Factory to create the Linked service and datasets and pipeline

Check below screenshot for **creating linked service** for our source i.e Azure SQL Database

Go to Monitor => Linked Service => New => search Azure SQL and select Azure SQL Database => Continue





New linked service

HTTP Learn more

ls_http_orders_dataset

Description

Connect via integration runtime * AutoResolveIntegrationRuntime

Base URL * https://files.cdn.thinkific.com

Information will be sent to the URL specified. Please ensure you trust the URL entered.

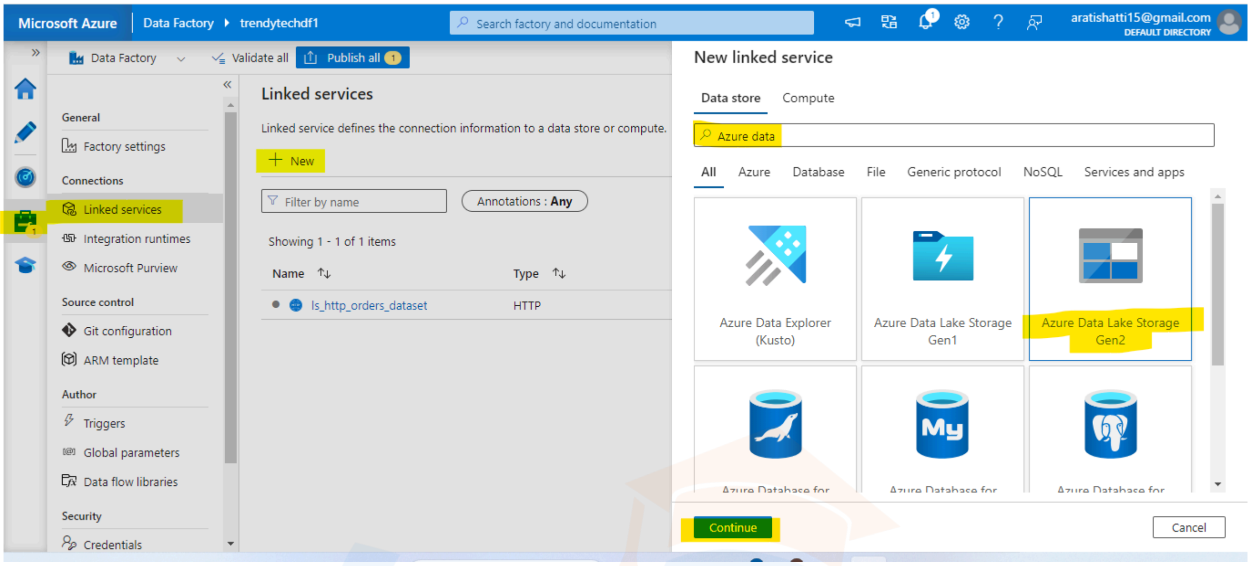
Server Certificate Validation ● Enable ○ Disable

Authentication type * Anonymous

Connection successful

Create a Linked Service for Sink (choose Data Lake Gen 2 connector) :

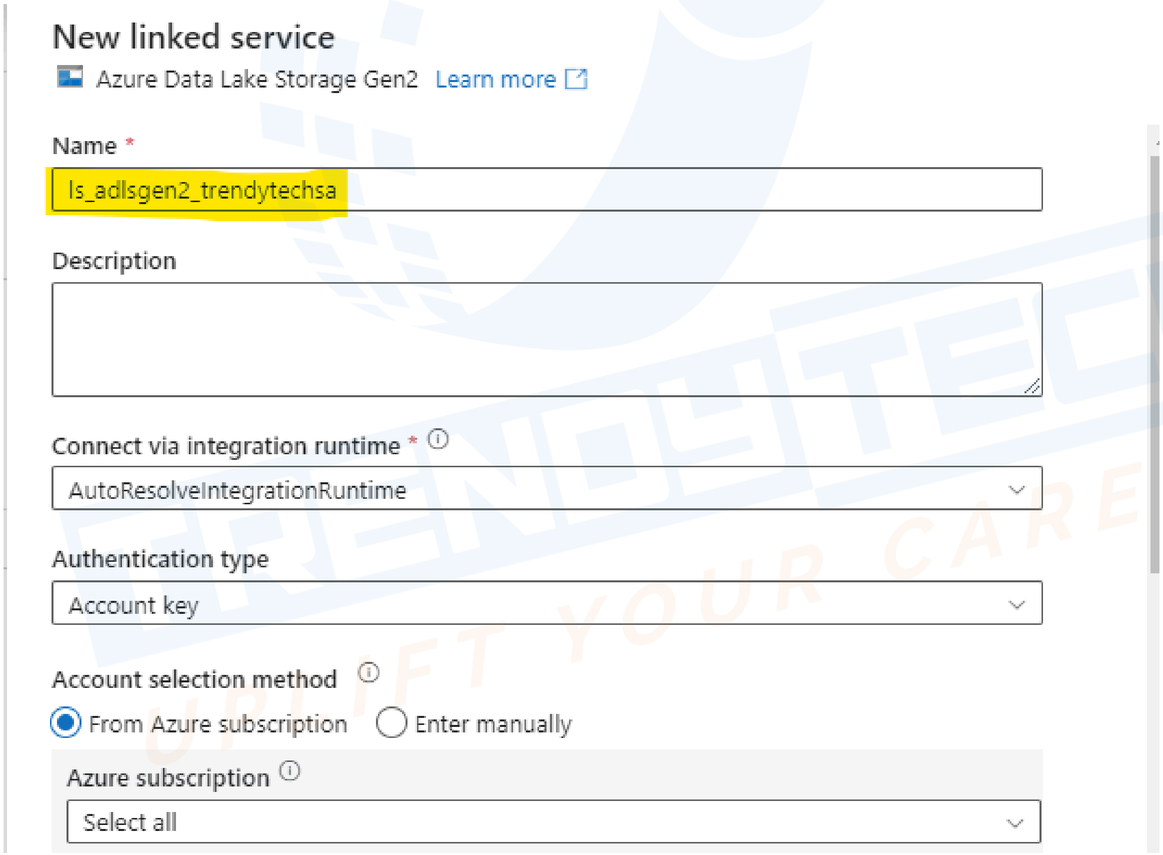
Check below screenshot for creating linked service for our sink i.e ADLS gen storage



Linked service defines the connection information to a data store or compute.

Name	Type
ls_http_orders_dataset	HTTP

Azure data



New linked service

Azure Data Lake Storage Gen2 Learn more

Name *

ls_adlsgen2_trendytechsa

Description

Connect via integration runtime *

AutoResolveIntegrationRuntime

Authentication type

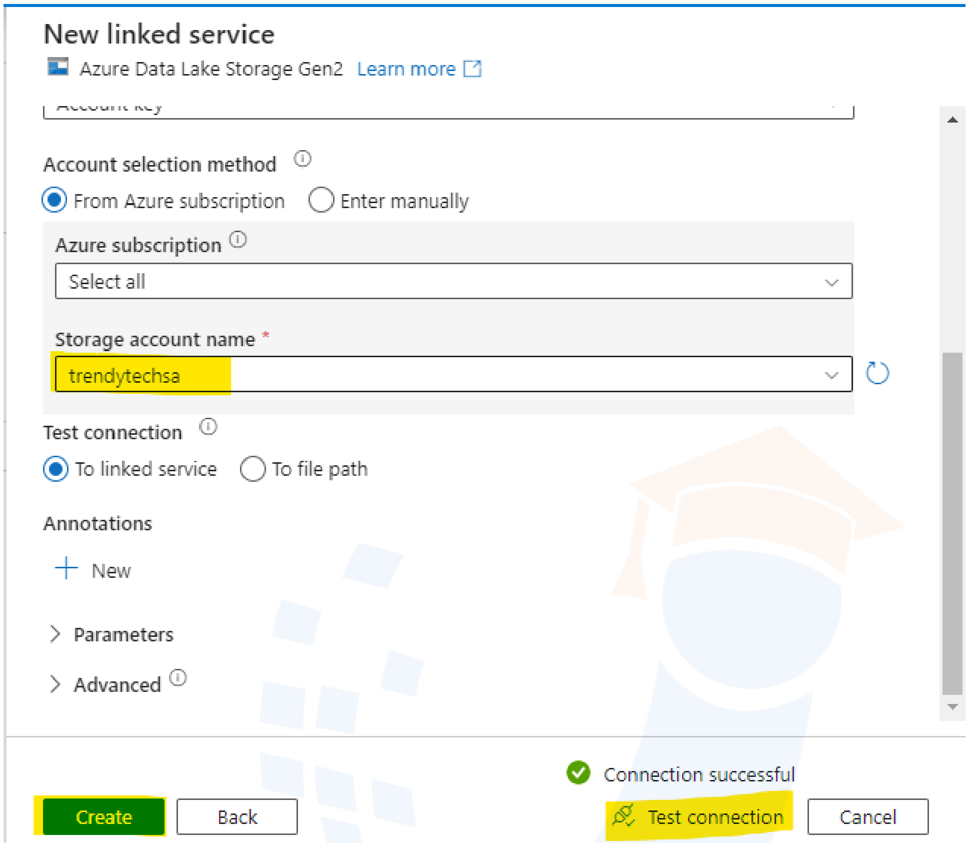
Account key

Account selection method

● From Azure subscription ○ Enter manually

Azure subscription

Select all



New linked service

Azure Data Lake Storage Gen2 Learn more

Account selection method

● From Azure subscription ○ Enter manually

Azure subscription: Select all

Storage account name *: trendytechsa

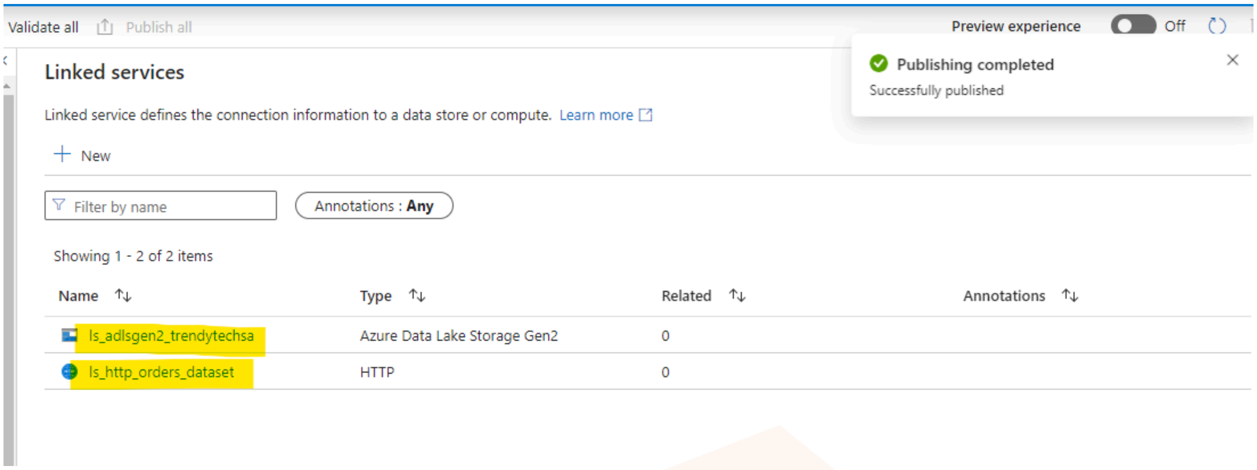
Test connection

● To linked service ○ To file path

Annotations

Connection successful

Note: After creating new linked services in Azure Data Factory, be sure to publish these changes to make them active and available for use in your data workflows.



Publishing completed

Successfully published

Linked services

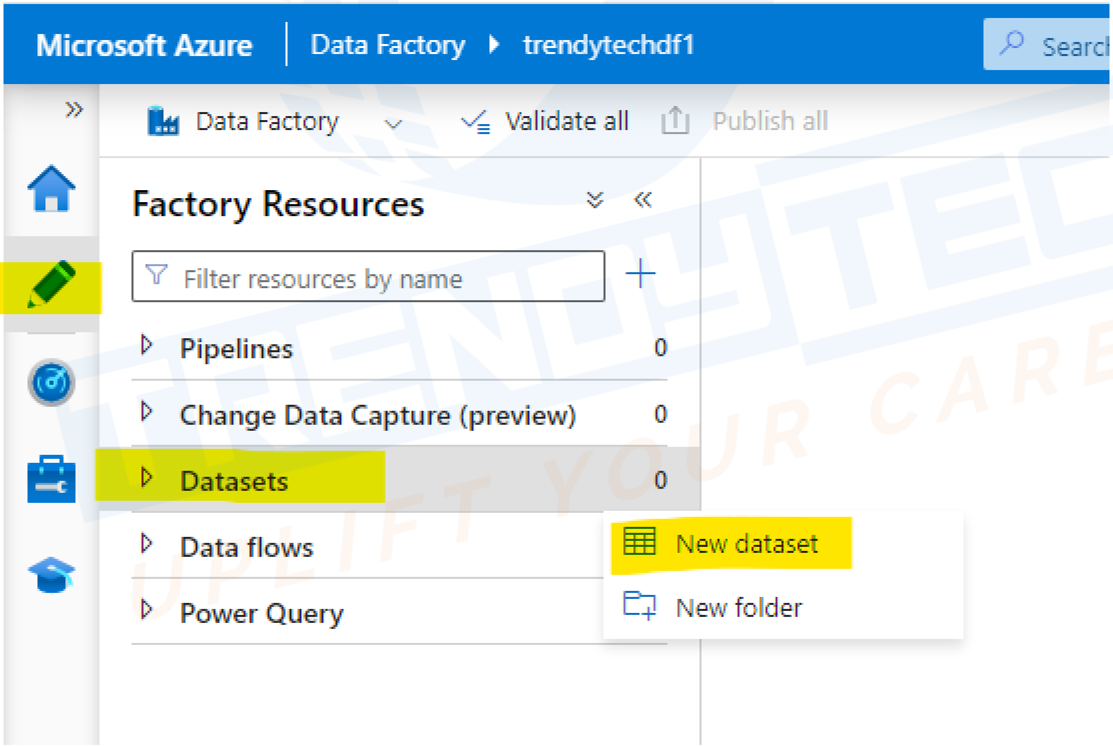
Linked service defines the connection information to a data store or compute. Learn more

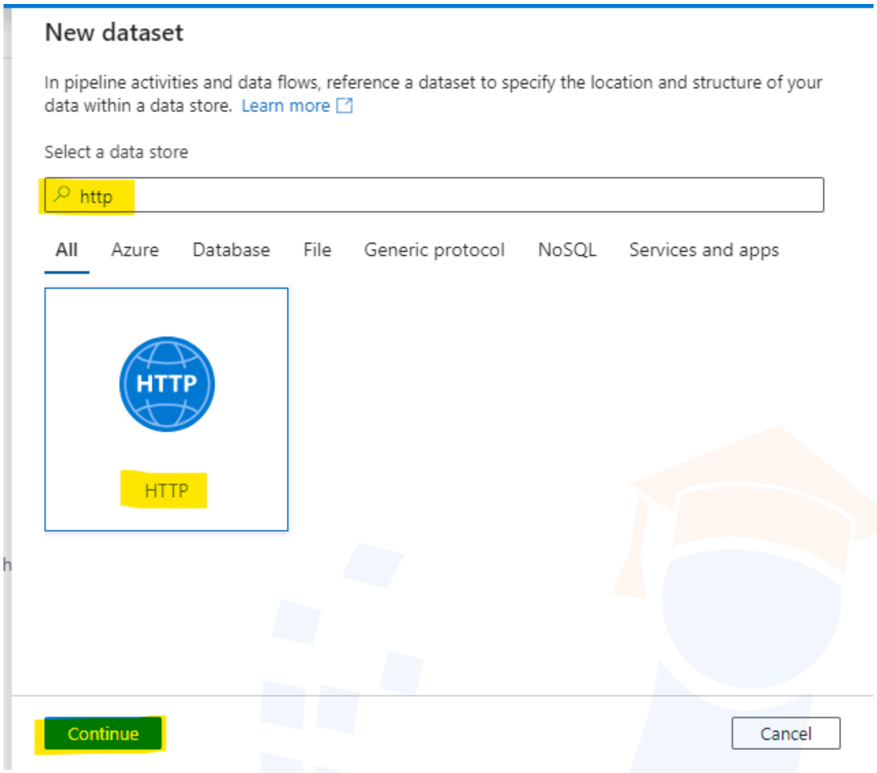
Showing 1 - 2 of 2 items

Name	Type	Related	Annotations
ls_adlsgen2_trendytechsa	Azure Data Lake Storage Gen2	0	
ls_http_orders_dataset	HTTP	0	

Create Dataset for Source (choose CSV format and also provide the relative URL as it is for the HTTP Linked Service)

To create dataset click on Author => Datasets => New datasets





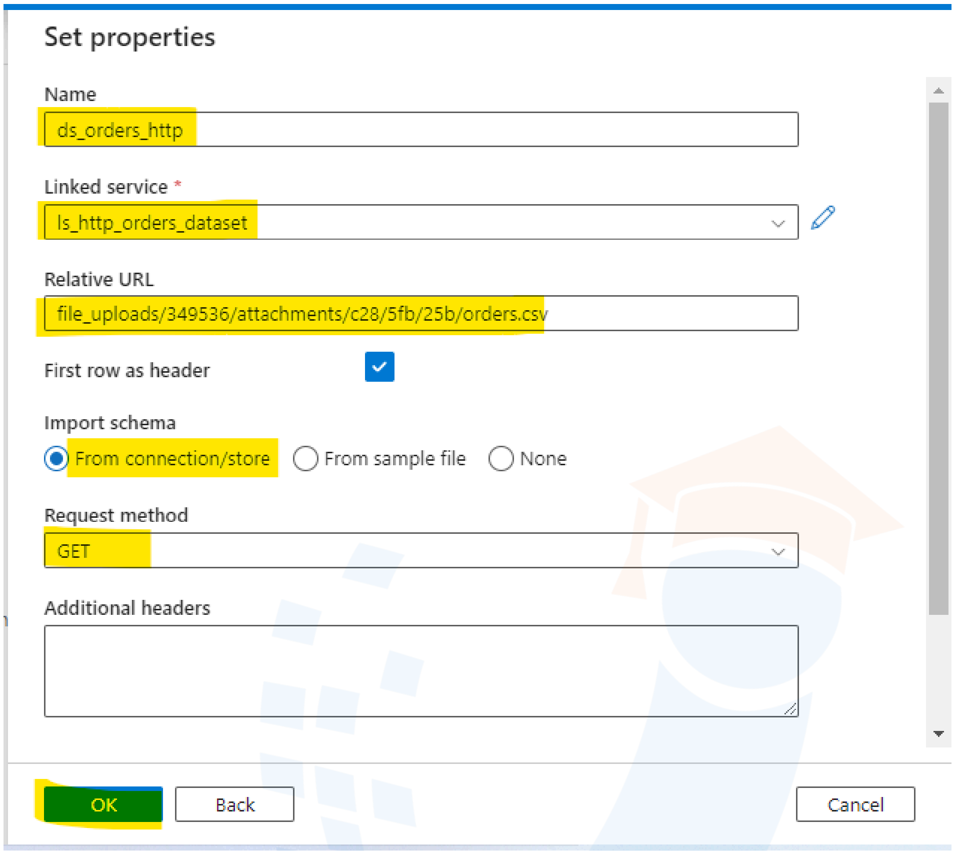
New dataset

In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. Learn more

Select a data store

http

BaseURL - https://files.cdn.thinkific.com
Relative URL - file_uploads/349536/attachments/c28/5fb/25b/orders.csv



Set properties

Name

ds_orders_http

Linked service *

ls_http_orders_dataset

Relative URL

file_uploads/349536/attachments/c28/5fb/25b/orders.csv

First row as header ☑

Import schema

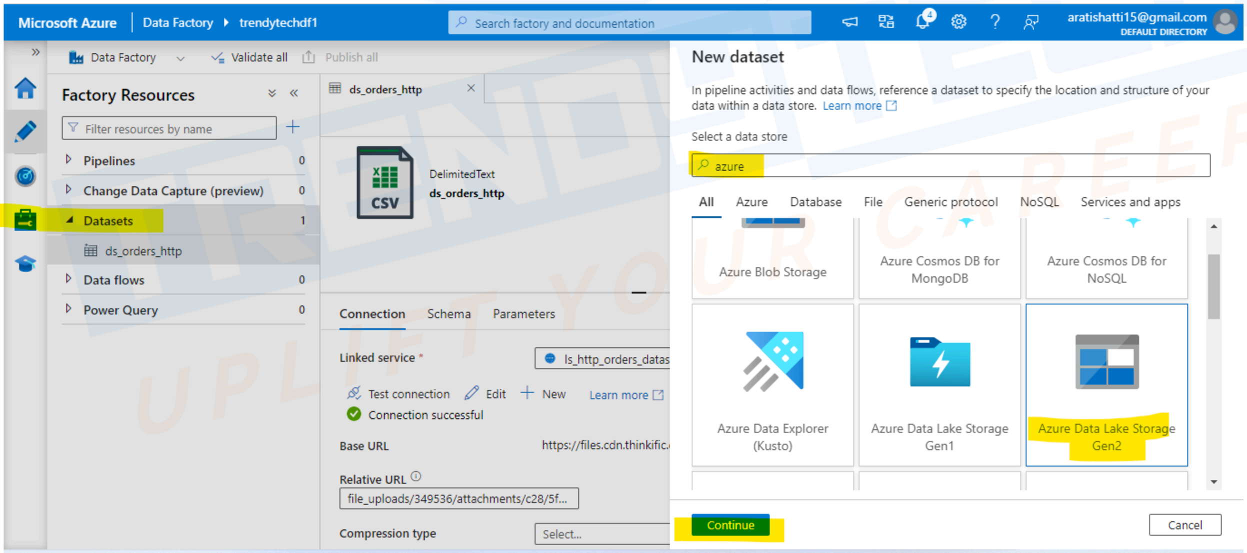
● From connection/store ○ From sample file ○ None

Request method

GET

Additional headers

Create Dataset for Sink :



Microsoft Azure | Data Factory ▸ trendytechdf1

aratishatti15@gmail.com
DEFAULT DIRECTORY

DelimitedText
ds_orders_http

Connection | Schema | Parameters

Linked service * ls_http_orders_datas

Test connection | Edit | New | Learn more

Connection successful

Base URL https://files.cdn.thinkific.

Relative URL file_uploads/349536/attachments/c28/5f...

Compression type Select...

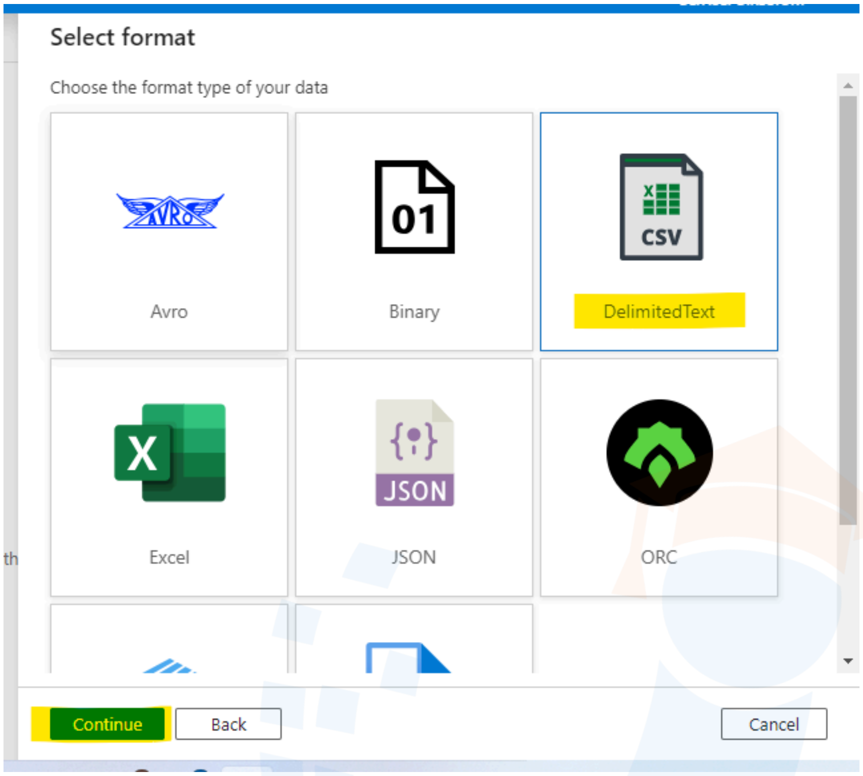
New dataset

In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. Learn more

Select a data store

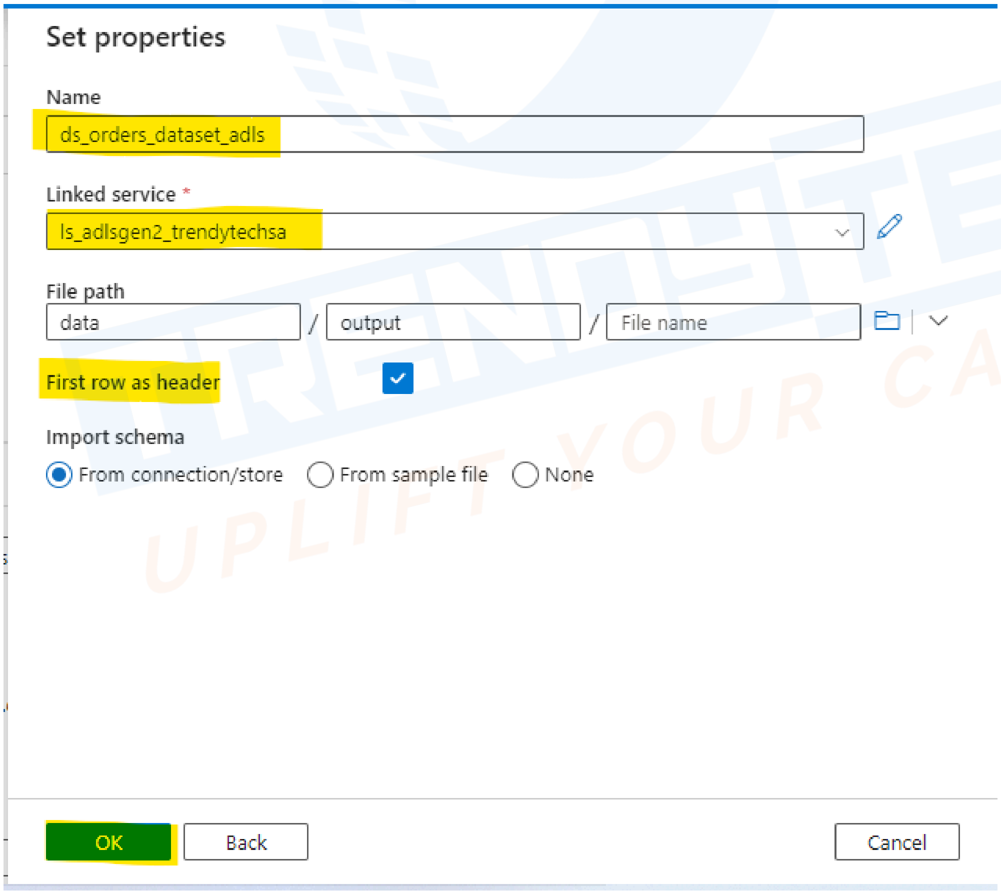
azure

All
Azure
Database
File
Generic protocol
NoSQL
Services and apps
Azure Blob Storage
Azure Cosmos DB for MongoDB
Azure Cosmos DB for NoSQL
Azure Data Explorer (Kusto)
Azure Data Lake Storage Gen1
Azure Data Lake Storage Gen2
Continue
Cancel



Select format

Choose the format type of your data



Set properties

Name: ds_orders_dataset_adls

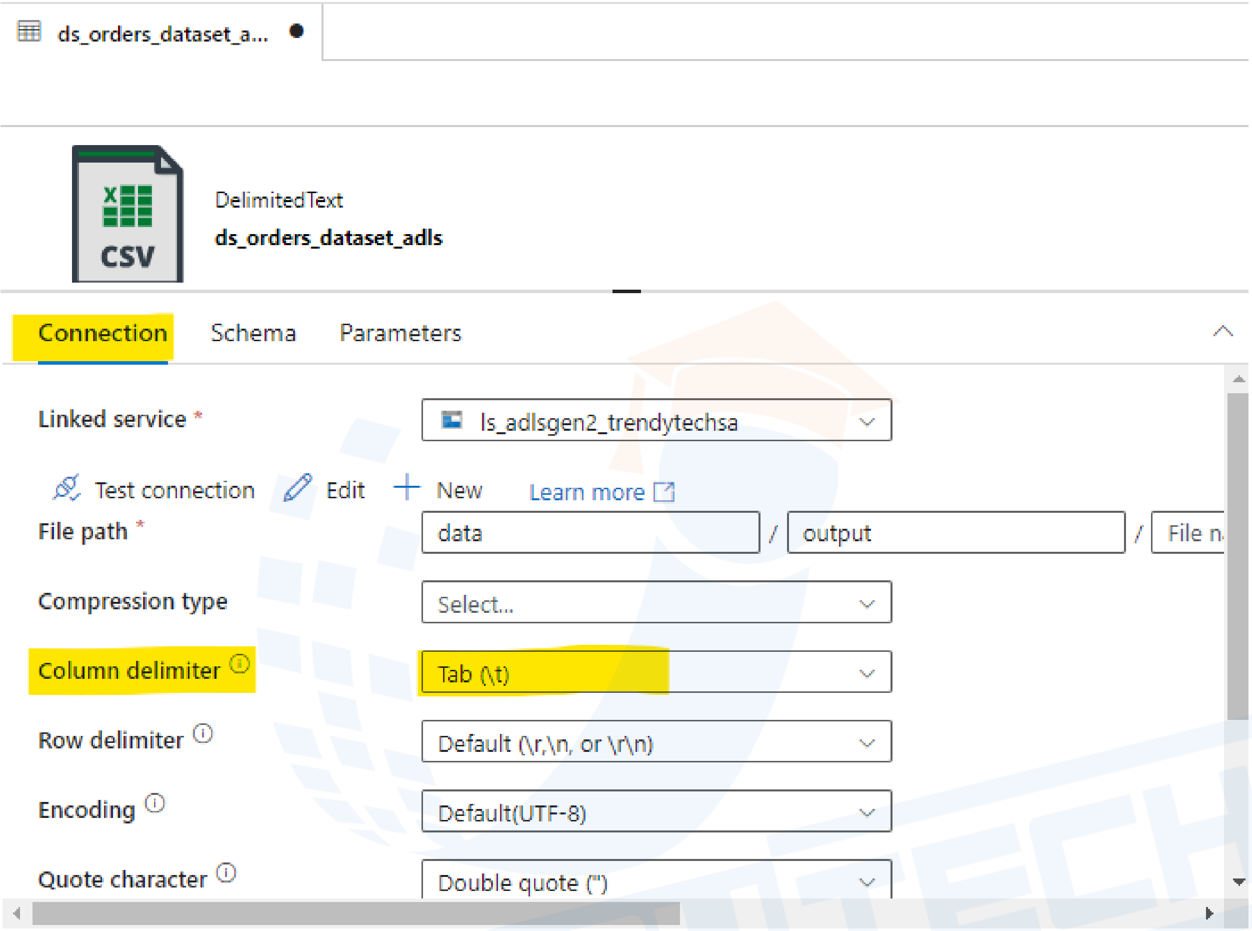
Linked service *: ls_adlsgen2_trendytechsa

File path: data / output / File name

First row as header ☑

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

Note: Here we will use separator as “Tab”



ds_orders_dataset_a... ●

DelimitedText
ds_orders_dataset_adls

Linked service * ls_adlsgen2_trendytechsa

File path * data / output / File n

Compression type: Select...

Column delimiter: Tab (\t)

Row delimiter: Default (\r,\n, or \r\n)

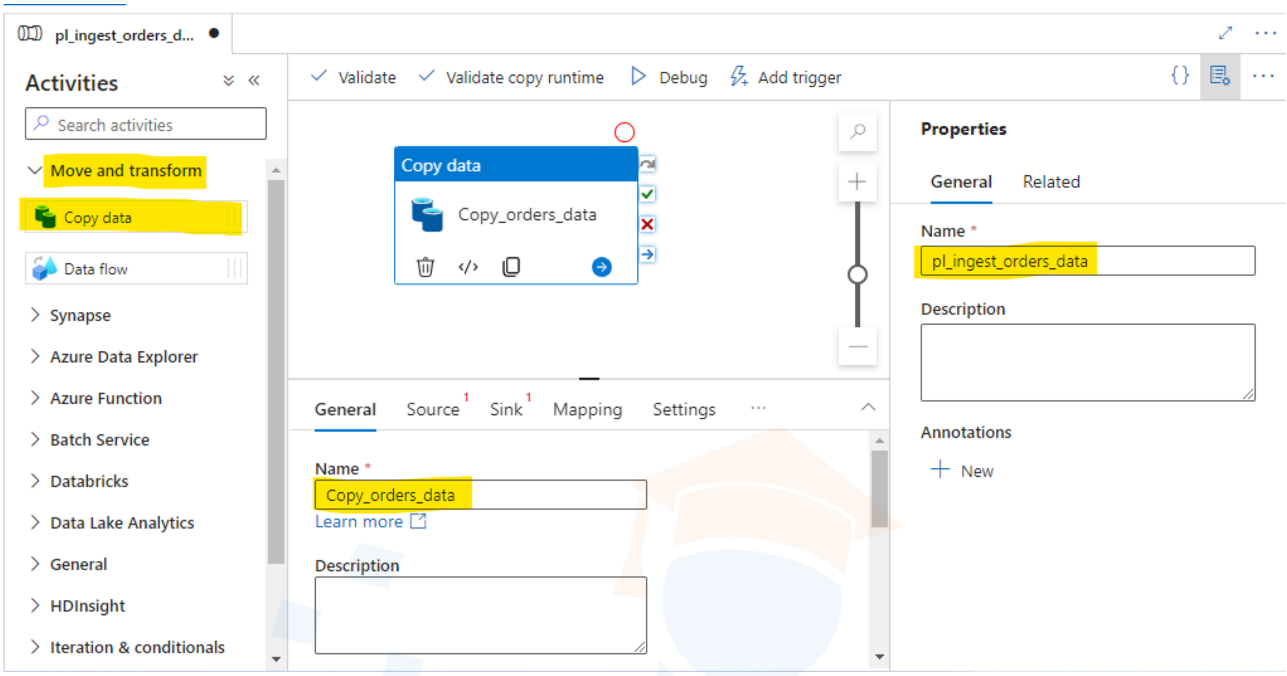
Encoding: Default(UTF-8)

Quote character: Double quote (")

Note: After creating new datasets in Azure Data Factory, be sure to publish these changes to make them active and available for use in your data workflows.

Create a Data Pipeline and Create a Copy activity within the Pipeline :

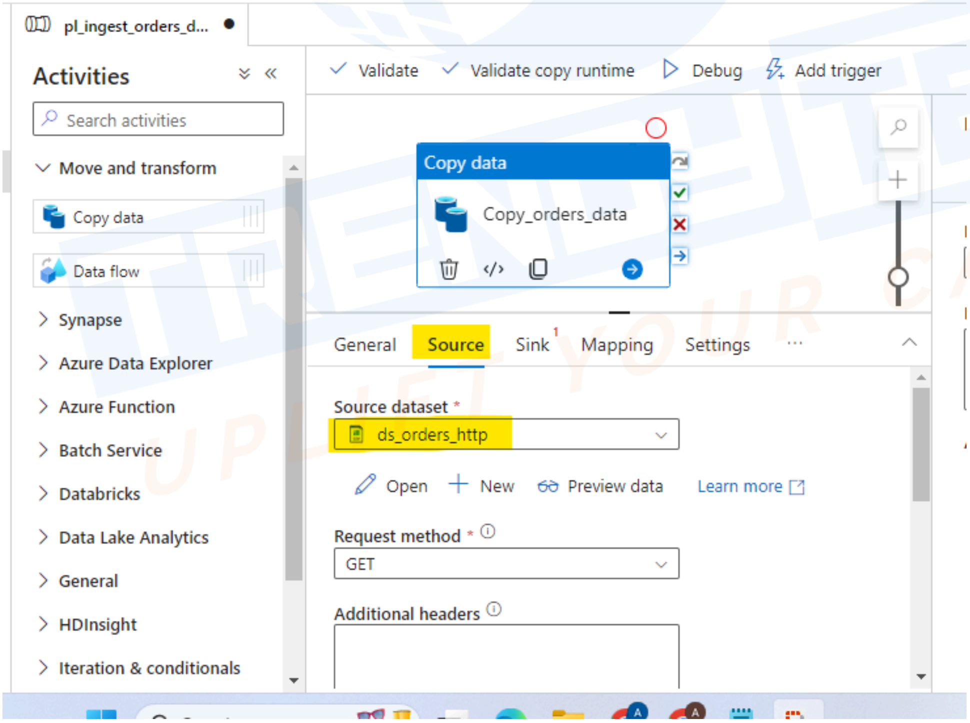
Now click on “**Move and transform**” and drag copy activity in the pipeline as shown below. Rename the pipeline copy data activity as shown in screenshot.



Name * Copy_orders_data

Properties — Name * pl_ingest_orders_data

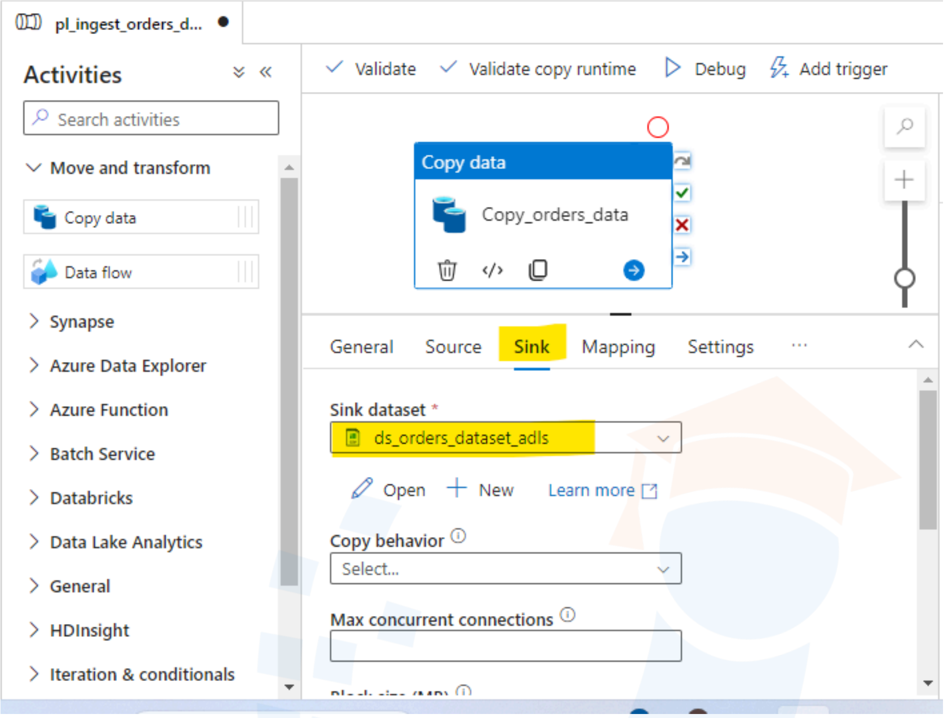
Now in source select dataset for course table in sql db here “ds_orders_http”



Source dataset * ds_orders_http

Request method * GET

Now in sink select dataset for sink here “ds_orders_dataset_adls”



Sink dataset *

ds_orders_dataset_adls

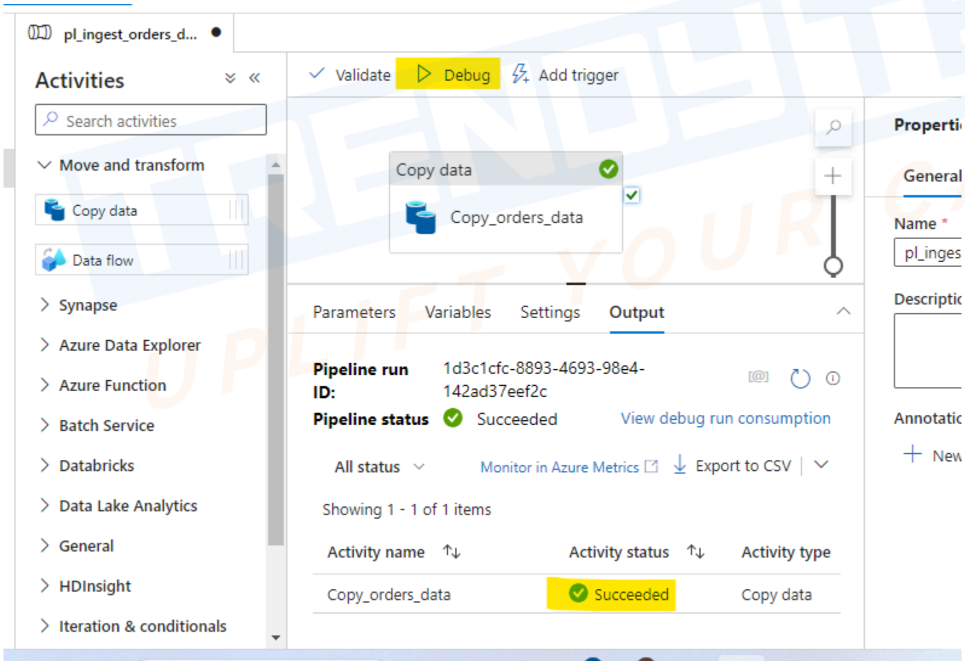
Open + New Learn more

Copy behavior

Select...

Max concurrent connections

Debug and validate the pipeline, and upon successful validation, proceed to publish the pipeline:



Pipeline run ID: 1d3c1cfc-8893-4693-98e4-142ad37eef2c

Pipeline status: Succeeded

View debug run consumption

All status

Monitor in Azure Metrics

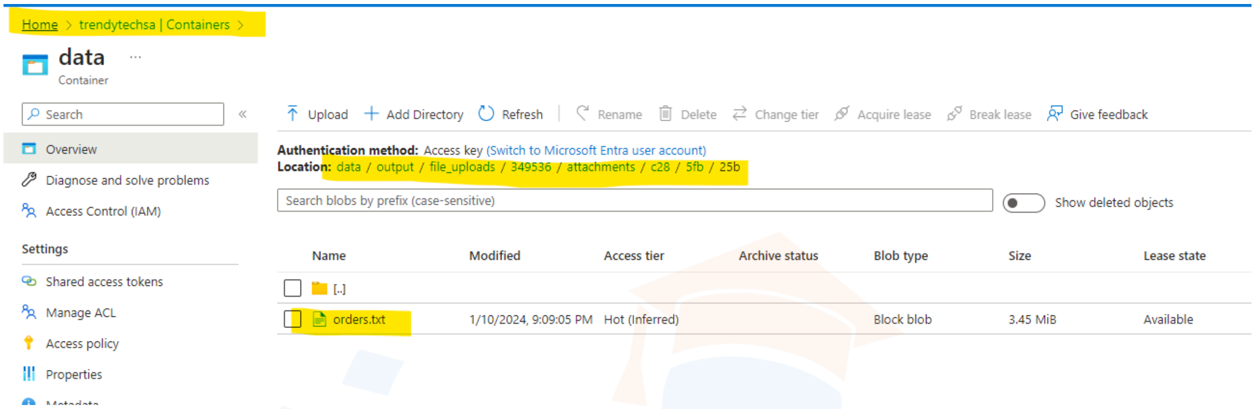
Export to CSV

Showing 1 - 1 of 1 items

Activity name	Activity status	Activity type
Copy_orders_data	Succeeded	Copy data

And if this is successful then publish the pipeline.

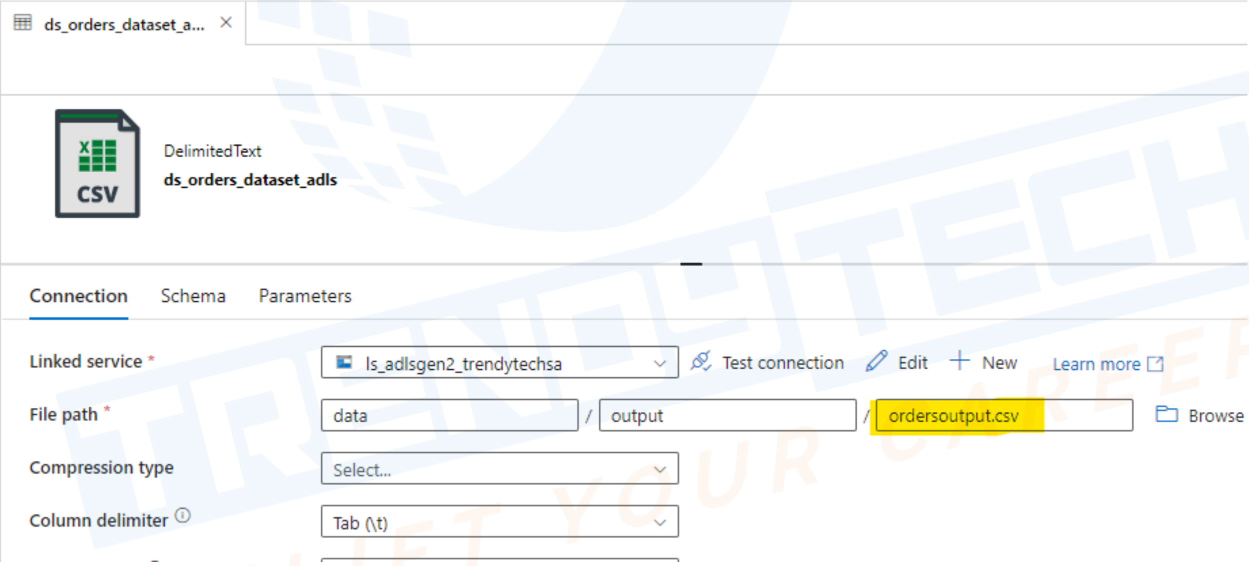
Note: After creating new pipelines in Azure Data Factory, be sure to publish these changes to make them active and available for use in your data workflows.



Authentication method: Access key (Switch to Microsoft Entra user account)
Location: data / output / file_uploads / 349536 / attachments / c28 / 5fb / 25b

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[..]						
orders.txt	1/10/2024, 9:09:05 PM	Hot (Inferred)		Block blob	3.45 MiB	Available

But here the file is not directly stored in output folder to avoid this we will edit the dataset “ds_orders_dataset_adls” and we will mention the file name as shown in below screenshot.



DelimitedText
ds_orders_dataset_adls

Linked service * ls_adlsgen2_trendytechsa

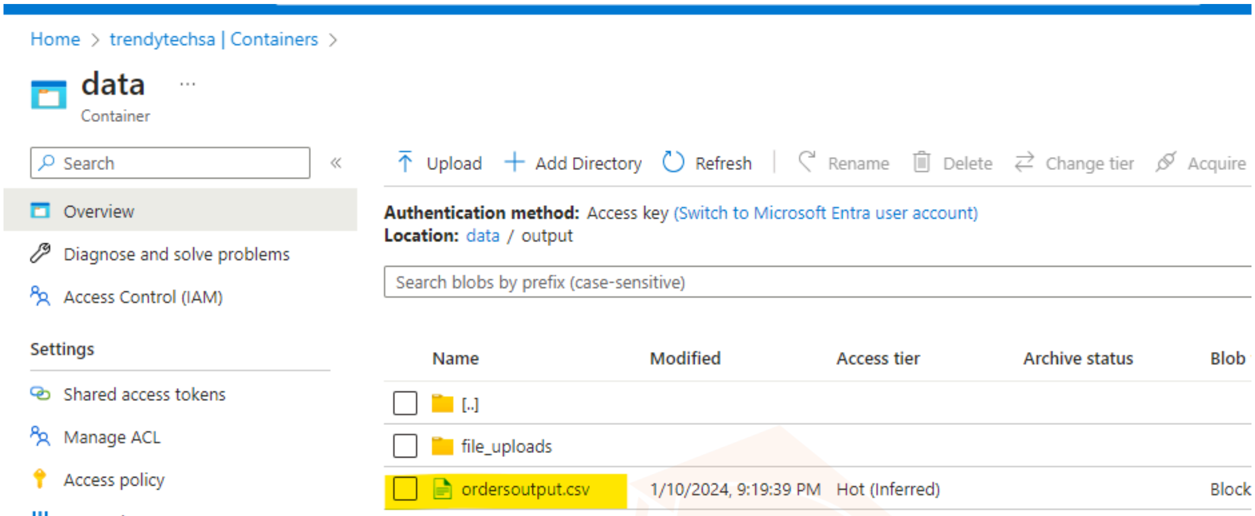
File path * data / output / ordersoutput.csv

Compression type Select...

Column delimiter Tab (\t)

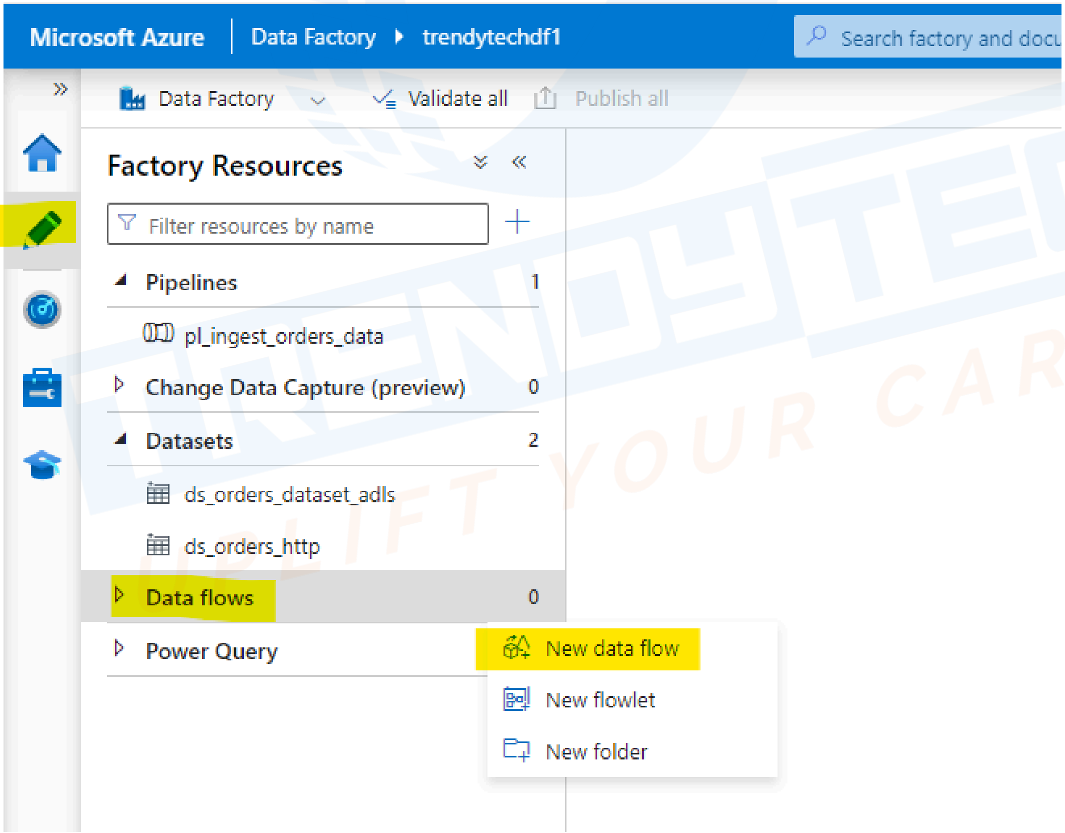
Note: Be sure to publish these changes to make them active and available for use in your data workflows.

Now again debug the pipeline and you will see the data is directly stored in “ordersoutput.csv” file refer attached screenshot.

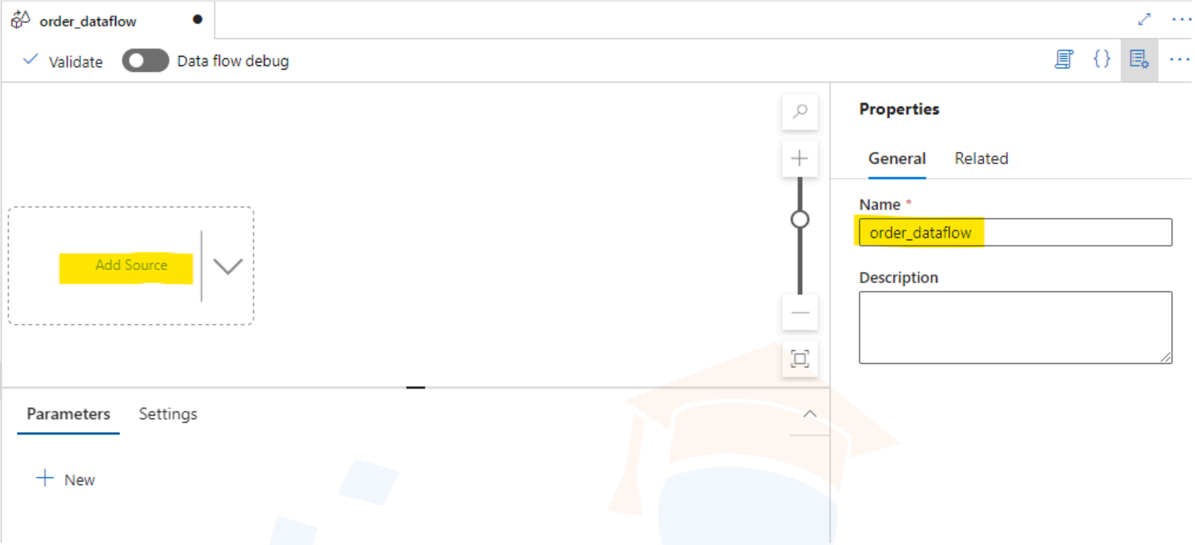


Authentication method: Access key (Switch to Microsoft Entra user account)
Location: data / output

	Name	Modified	Access tier	Archive status	Blob
□	[..]				
□	file_uploads				
□	ordersoutput.csv	1/10/2024, 9:19:39 PM	Hot (Inferred)		Block

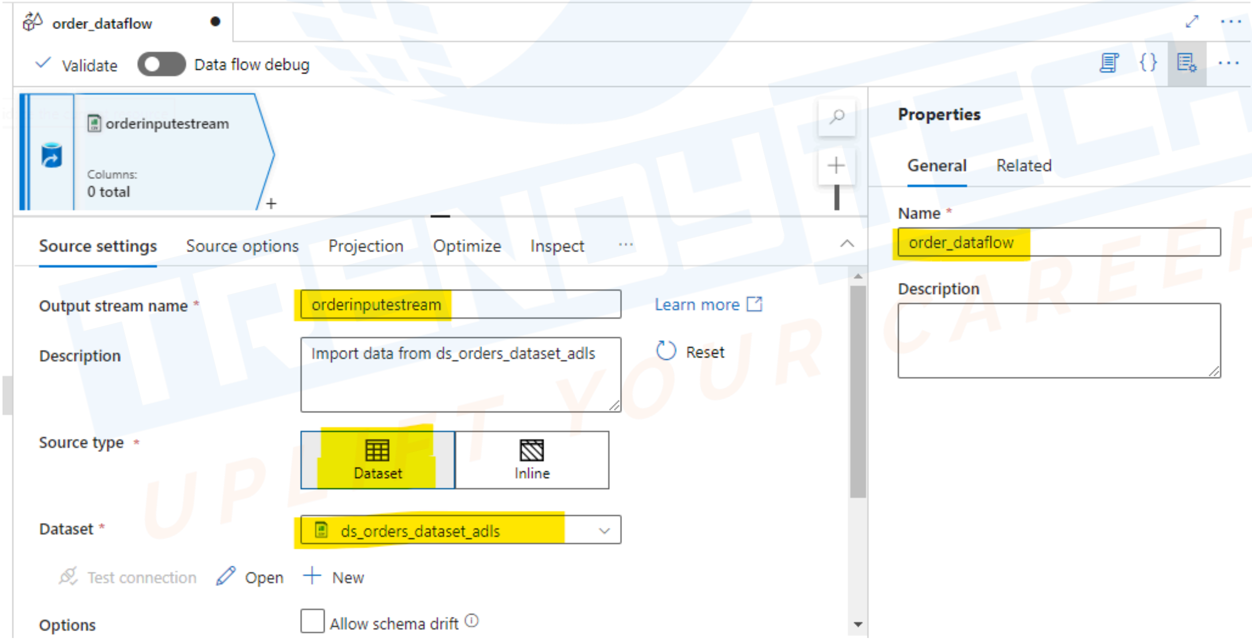


Create a data flow and click on the option "Add Source"



Name * order_dataflow

Now the **source for this data flow** will be our **"ordersoutput.csv"**dataset so we will mention the database for it i.e **"ds_orders_dataset_adls"**



Output stream name * orderinputestream

Description: Import data from ds_orders_dataset_adls

Source type * Dataset

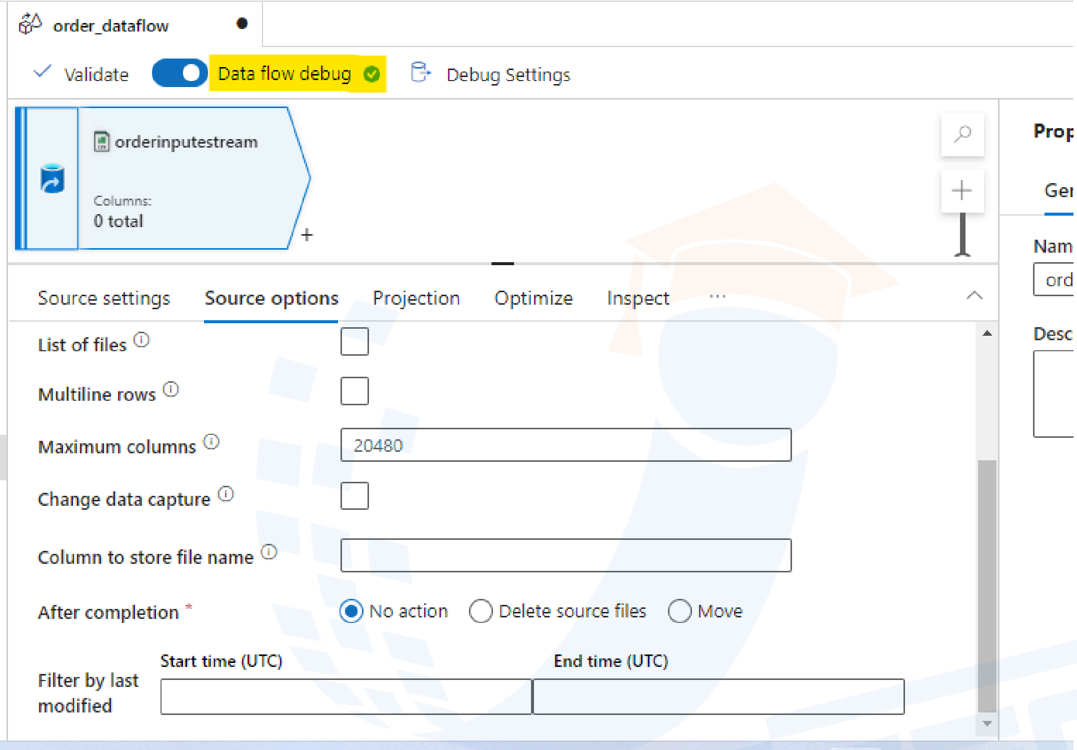
Dataset * ds_orders_dataset_adls

Options: □ Allow schema drift

Name * order_dataflow

Note : Data Preview is possible if a spark cluster is turned on. For development purposes, Debug mode can be used to preview the data after small transformation changes are made.

Refer below screenshot.



List of files □

Multiline rows □

Maximum columns 20480

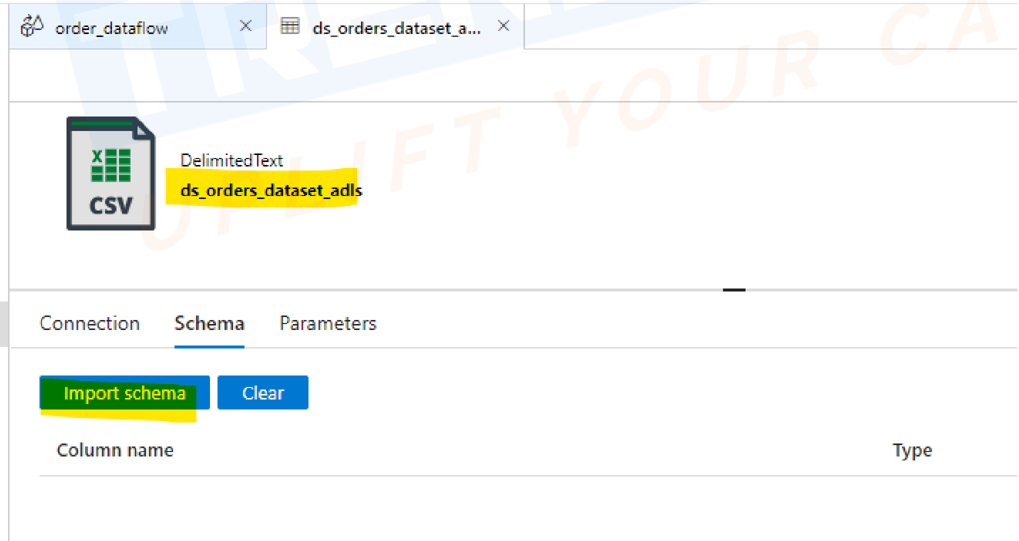
Change data capture □

Column to store file name

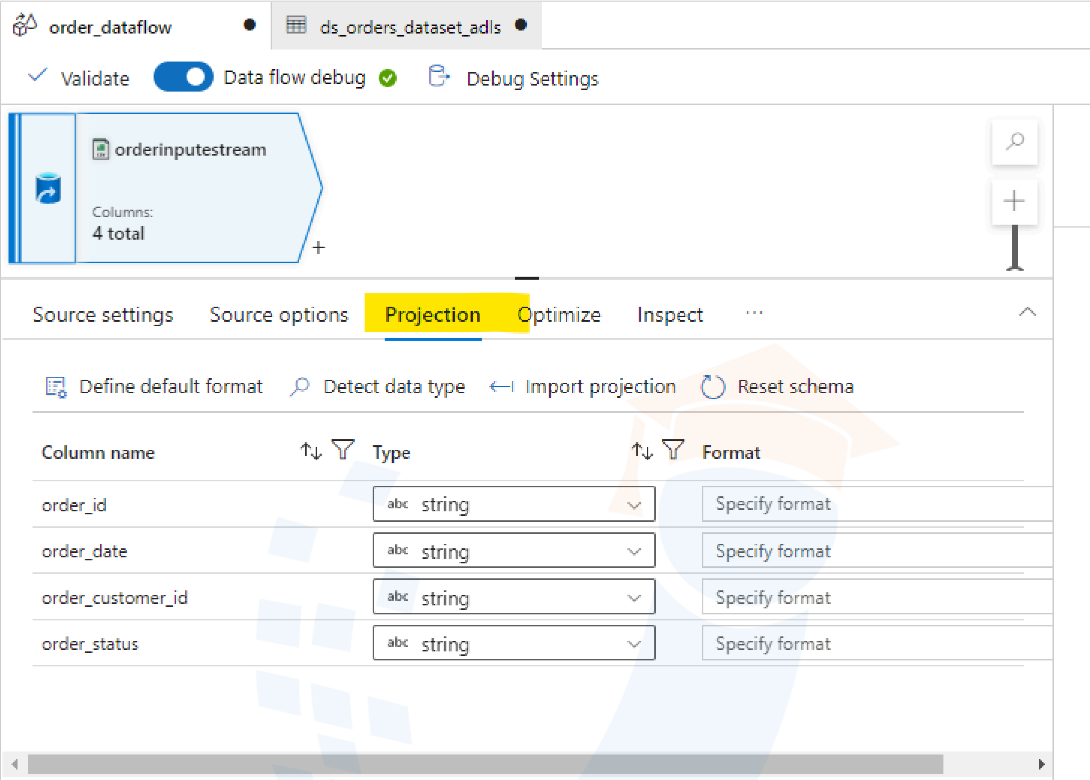
After completion * ● No action ○ Delete source files ○ Move

Filter by last modified — Start time (UTC) — End time (UTC)

Note: If you are not able to see the projection, you have to import the schema for the dataset “ds_orders_dataset_adls” .

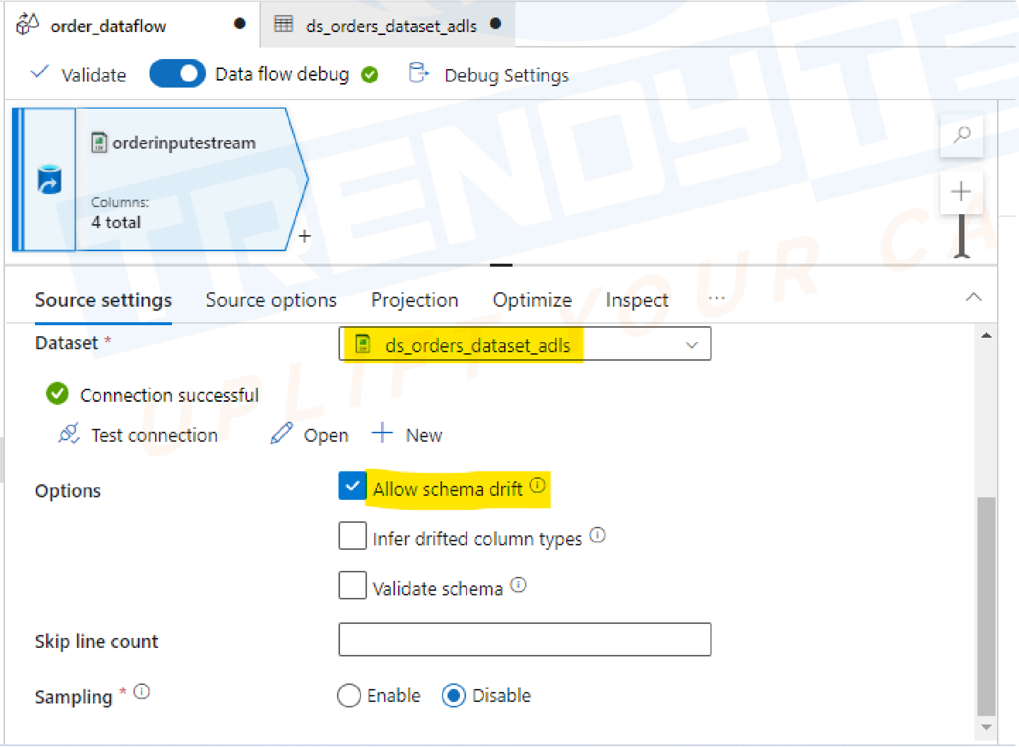


Now you can see the projections as shown below.



Column name	Type	Format
order_id	abc string	Specify format
order_date	abc string	Specify format
order_customer_id	abc string	Specify format
order_status	abc string	Specify format

Also select “Allow Schema Drift”



Dataset * ds_orders_dataset_adls

Connection successful

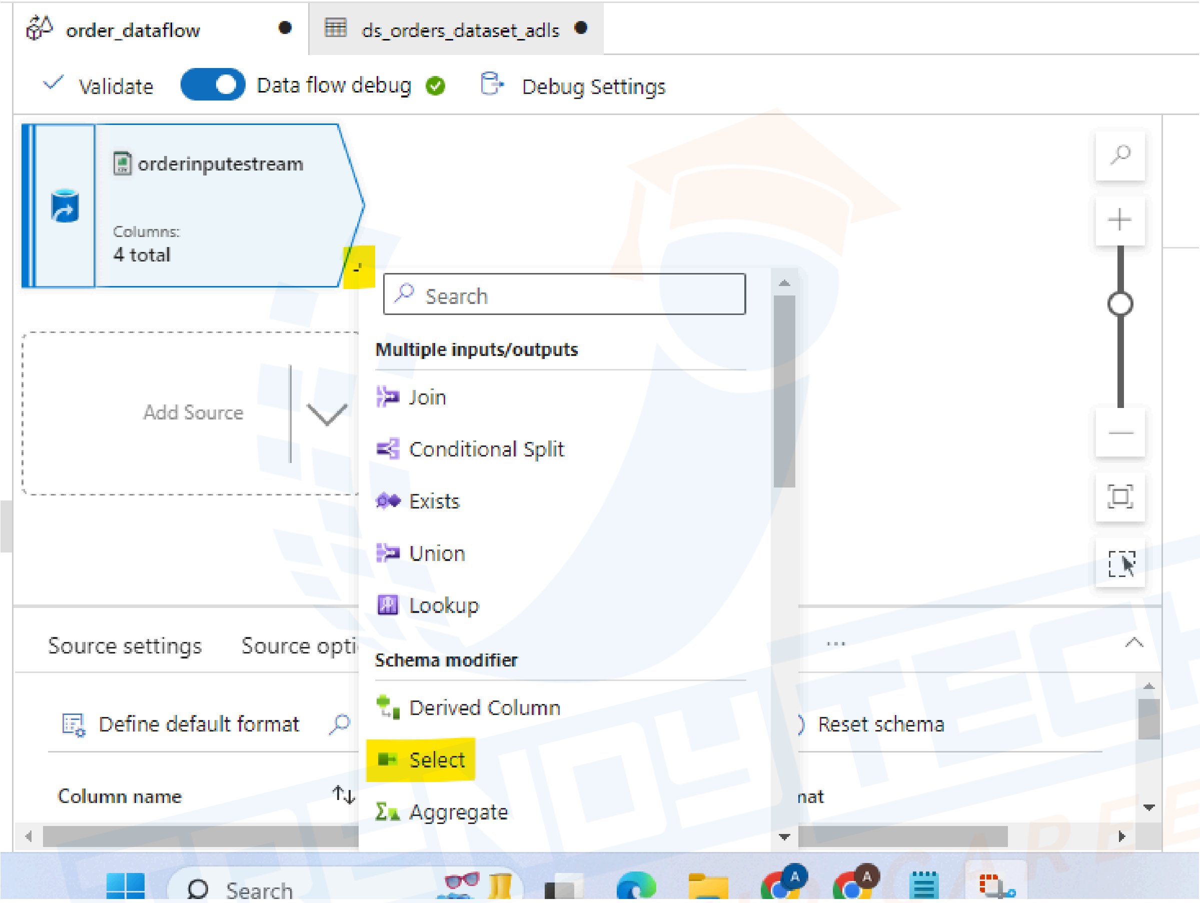
Options
☑ Allow schema drift
☐ Infer drifted column types
☐ Validate schema

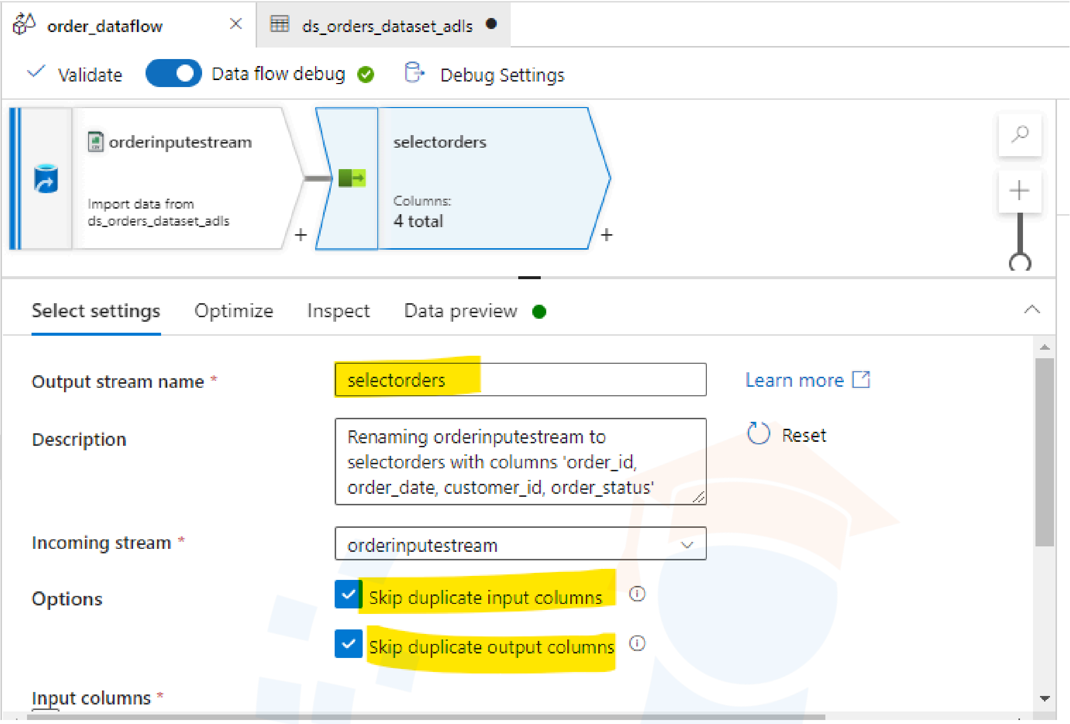
Skip line count

Sampling * ○ Enable ● Disable

Perform some basic transformations on this

To Remove order_date column and Rename order_customer_id to customer_id click on "+" we will use "select transformation" refer below screenshot





Output stream name * selectorders

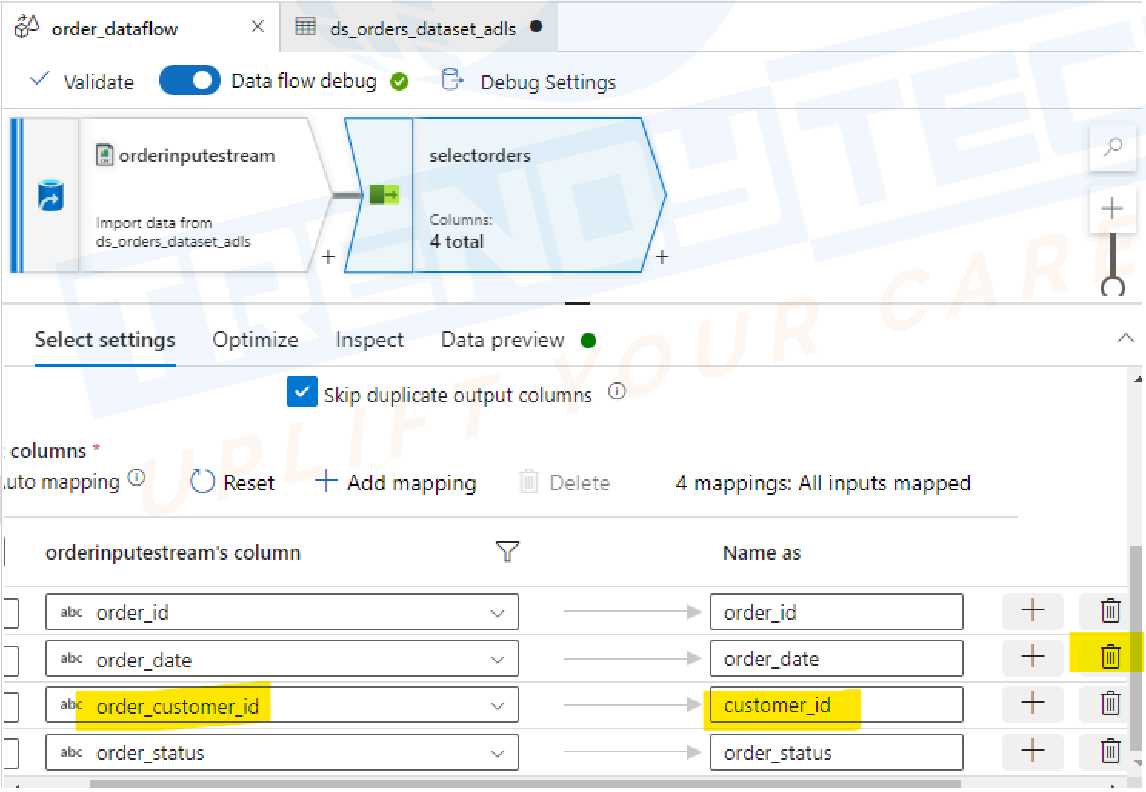
Description: Renaming orderinputestream to selectorders with columns 'order_id, order_date, customer_id, order_status'

Incoming stream * orderinputestream

Options
☑ Skip duplicate input columns
☑ Skip duplicate output columns

Input columns *

Now click on **delete option to remove order_date column** and **rename order_customer_id to customer_id** check highlighted options.

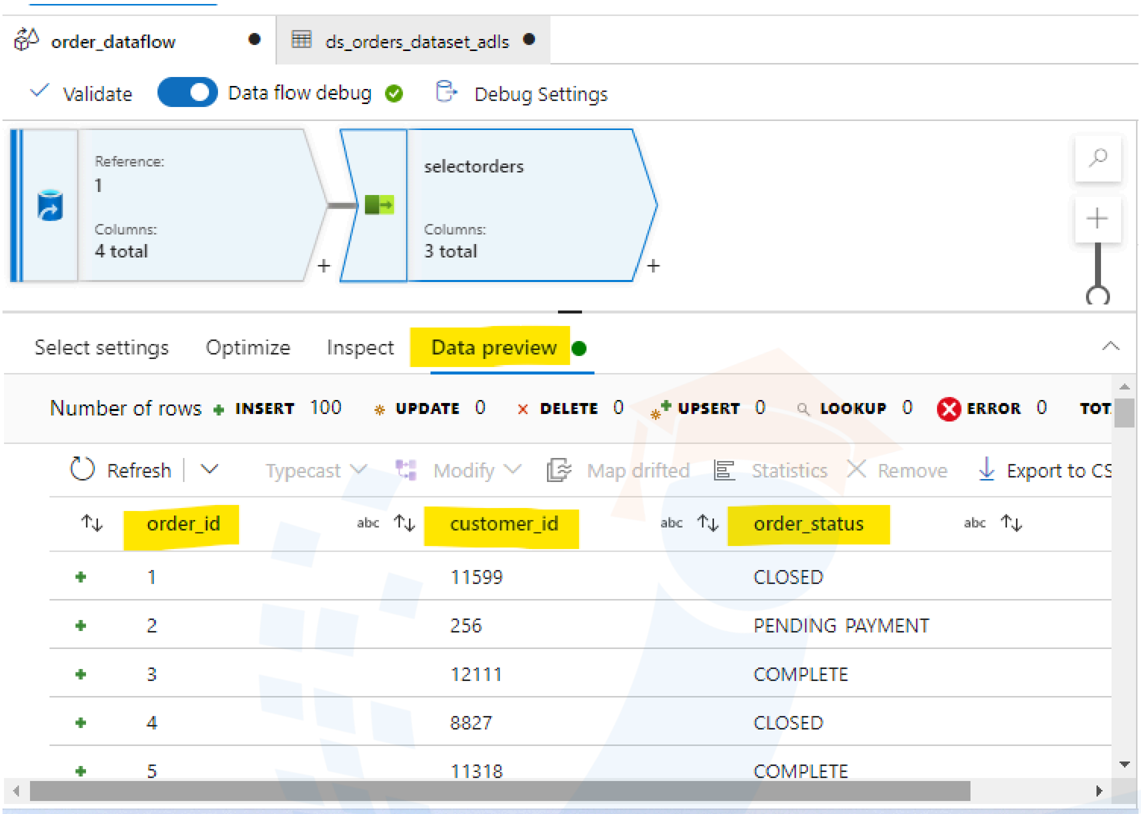


☑ Skip duplicate output columns

columns *
uto mapping · Reset · Add mapping · Delete · 4 mappings: All inputs mapped

orderinputestream's column	Name as
abc order_id	order_id
abc order_date	order_date
abc order_customer_id	customer_id
abc order_status	order_status

After this click on “Data Preview” and refresh it to see all the changes.



order_dataflow ● | ds_orders_dataset_adls ●

✓ Validate | Data flow debug | Debug Settings

Reference: 1 | Columns: 4 total

selectorders | Columns: 3 total

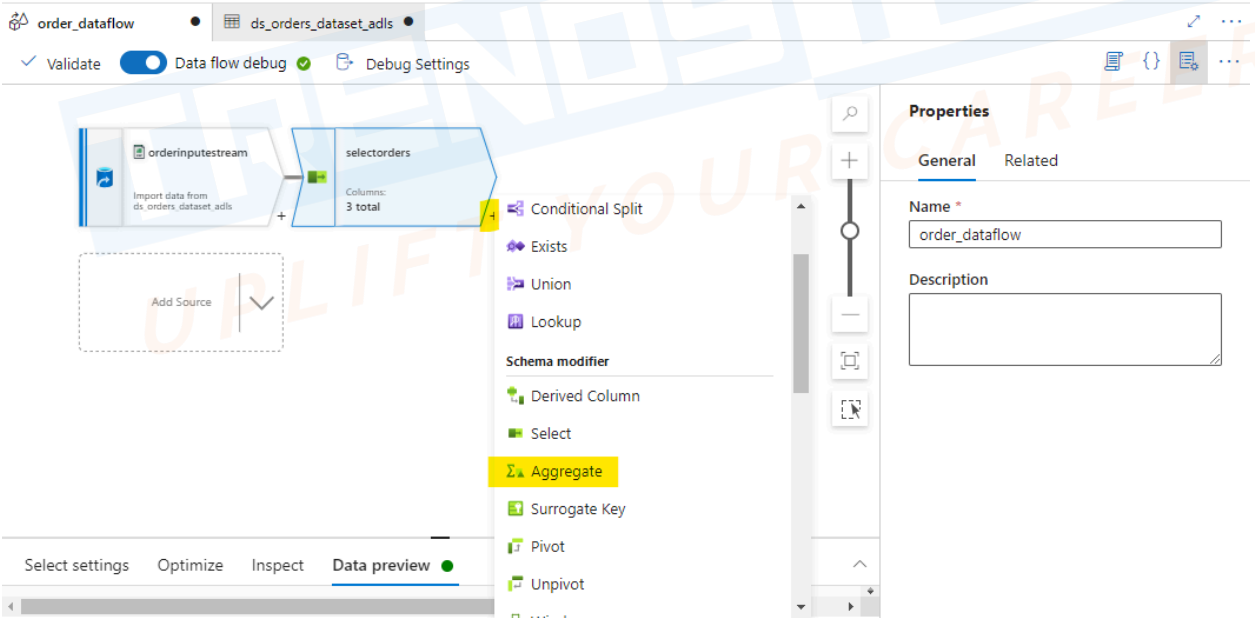
Select settings | Optimize | Inspect | Data preview ●

Number of rows INSERT 100 | UPDATE 0 | DELETE 0 | UPSERT 0 | LOOKUP 0 | ERROR 0 | TOT

Refresh | Typecast | Modify | Map drifted | Statistics | Remove | Export to CS

	order_id	customer_id	order_status
+	1	11599	CLOSED
+	2	256	PENDING_PAYMENT
+	3	12111	COMPLETE
+	4	8827	CLOSED
+	5	11318	COMPLETE

Now click on **“+”** we will use **“aggregate transformation”** refer below screenshot



order_dataflow ● | ds_orders_dataset_adls ●

✓ Validate | Data flow debug | Debug Settings

orderinputestream: Import data from ds_orders_dataset_adls

selectorders: Columns: 3 total

Add Source

Properties

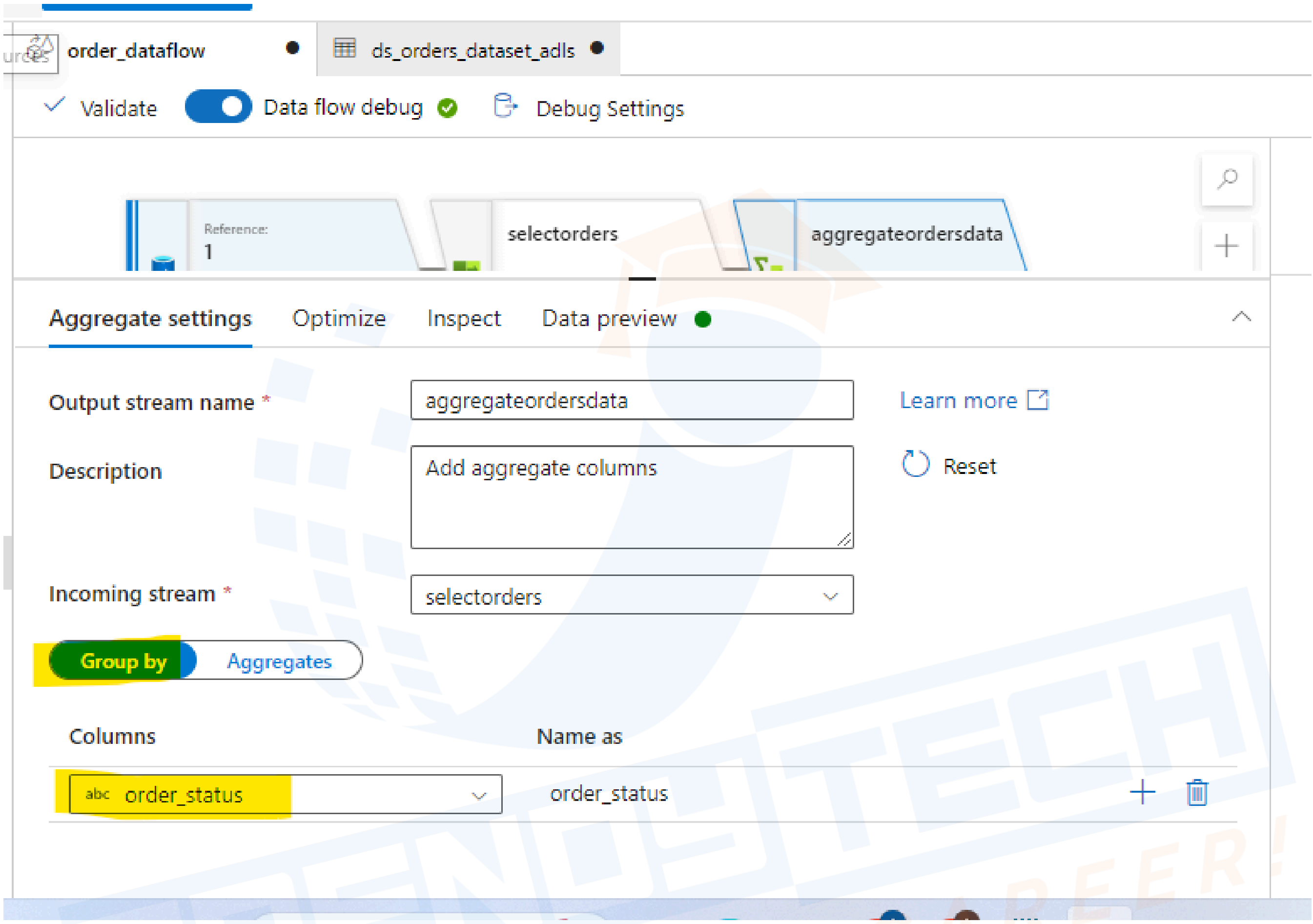
General | Related

Name * order_dataflow

Description

Select settings | Optimize | Inspect | Data preview ●

Now select the **group by** the “**order_status**” column as shown below.



Output stream name * aggregateordersdata

Description Add aggregate columns

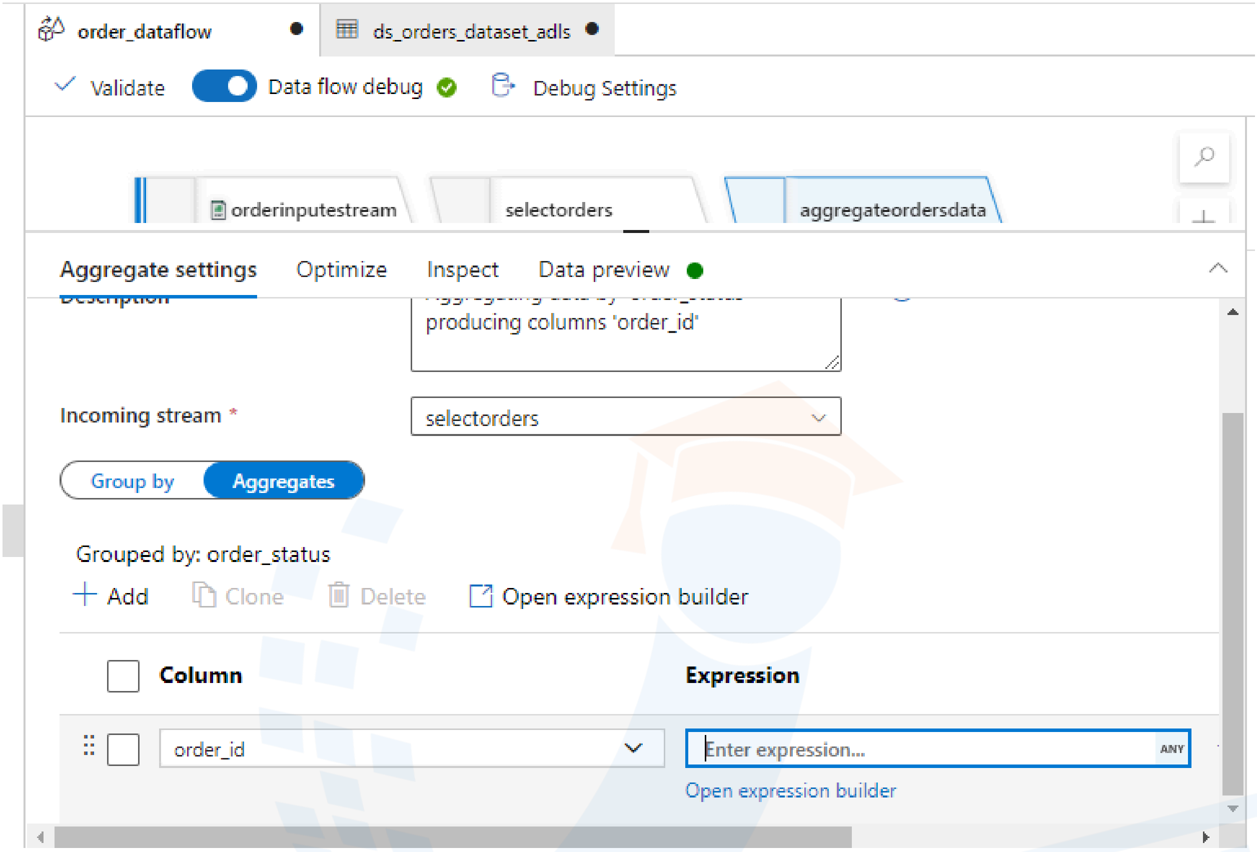
Incoming stream * selectorders

Columns	Name as
abc order_status	order_status

To calculate the count of each order status :

Click on **“Aggregates”** mention **order_id** column and click on **“Open expression builder”**

.



Description: producing columns 'order_id'

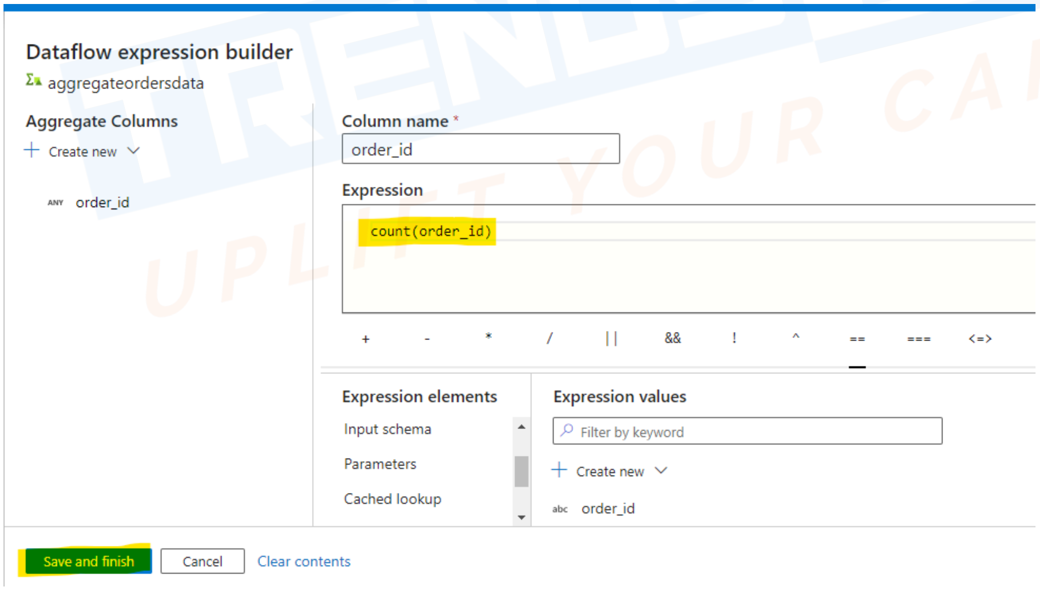
Incoming stream *: selectorders

Group by | Aggregates

Grouped by: order_status

Column	Expression
order_id	Enter expression...

Now mention **count(order_id) in “Expression”** refer attached screenshot and click **“save and finish”**

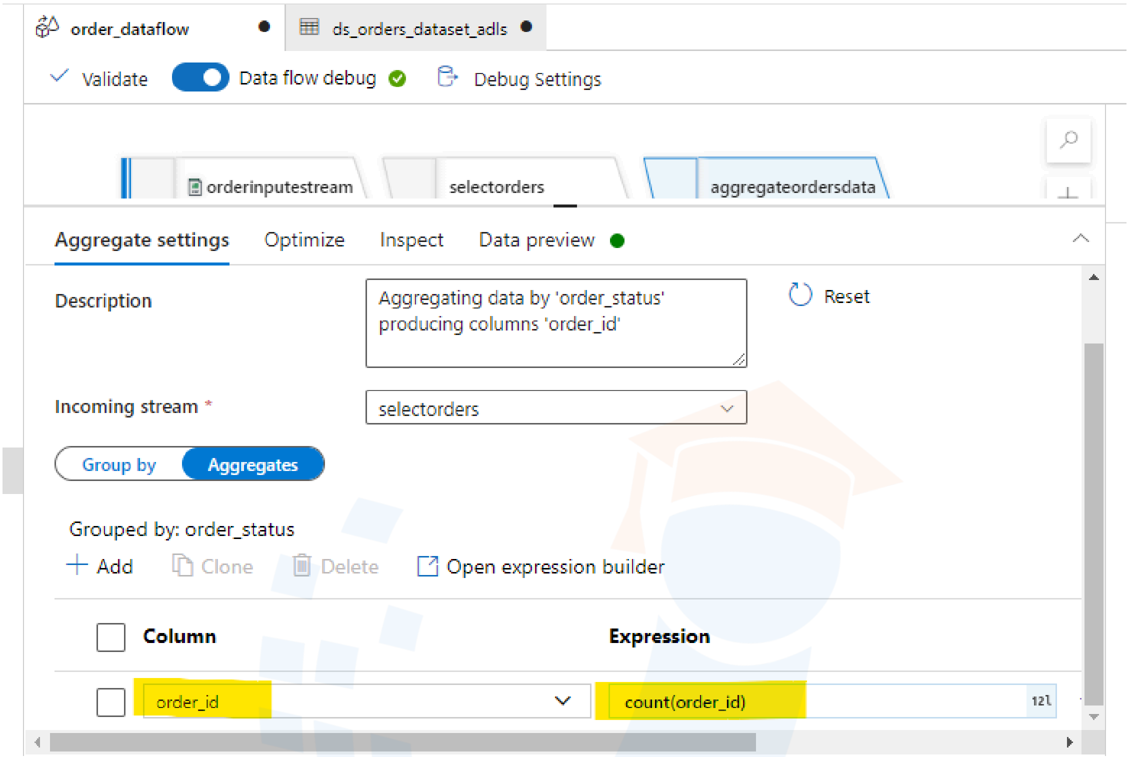

Dataflow expression builder

aggregateordersdata

Column name *: order_id

Expression:

```
count(order_id)
```



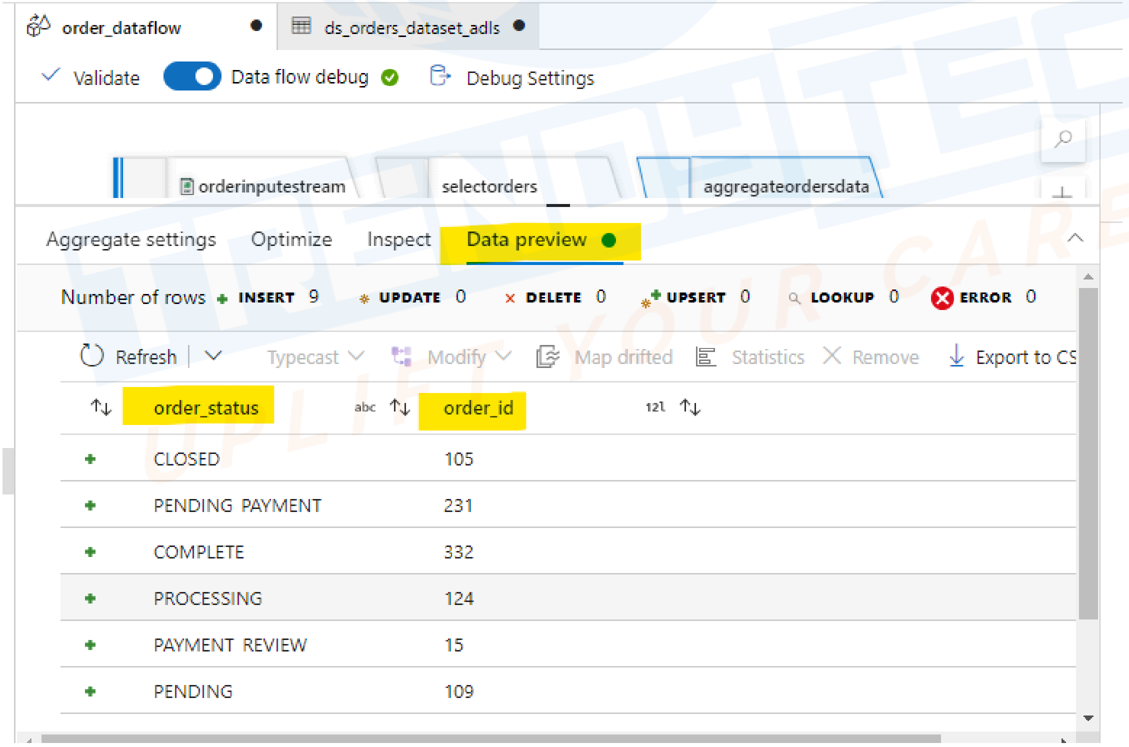
Description: Aggregating data by 'order_status' producing columns 'order_id'

Incoming stream *: selectorders

Grouped by: order_status

Column	Expression
order_id	count(order_id)

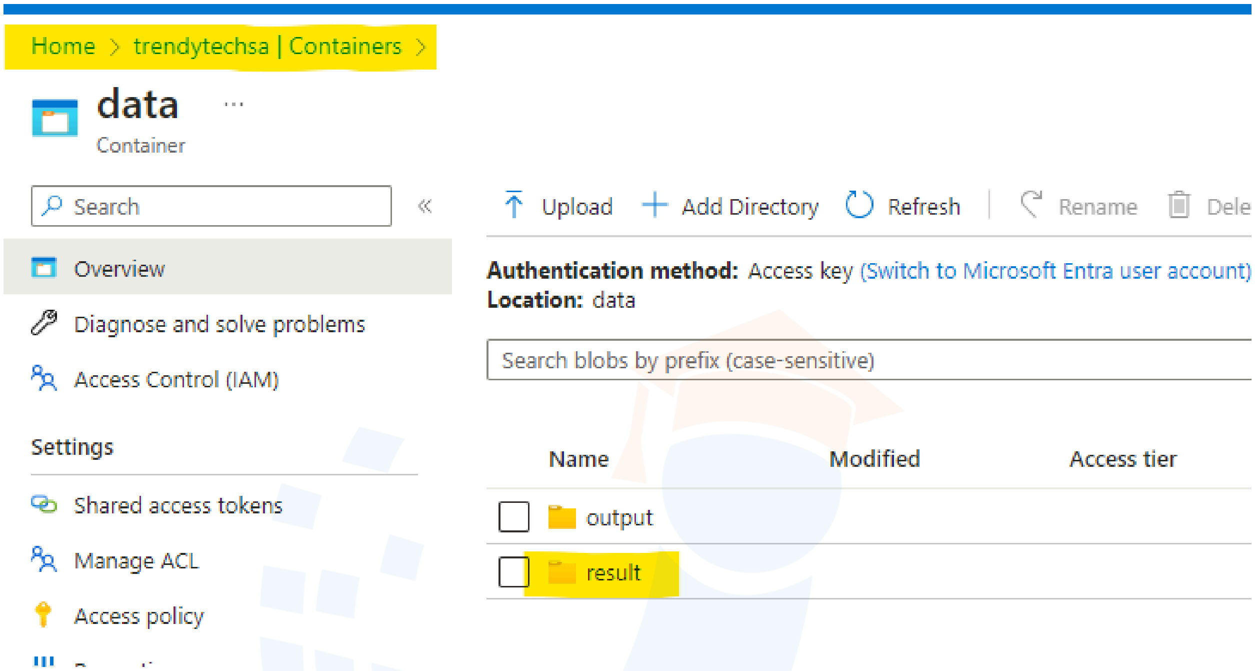
After this click on “Data Preview” and refresh it to see all the changes.



Number of rows + INSERT 9 UPDATE 0 × DELETE 0 UPSERT 0 LOOKUP 0 ERROR 0

	order_status	order_id
+	CLOSED	105
+	PENDING_PAYMENT	231
+	COMPLETE	332
+	PROCESSING	124
+	PAYMENT_REVIEW	15
+	PENDING	109

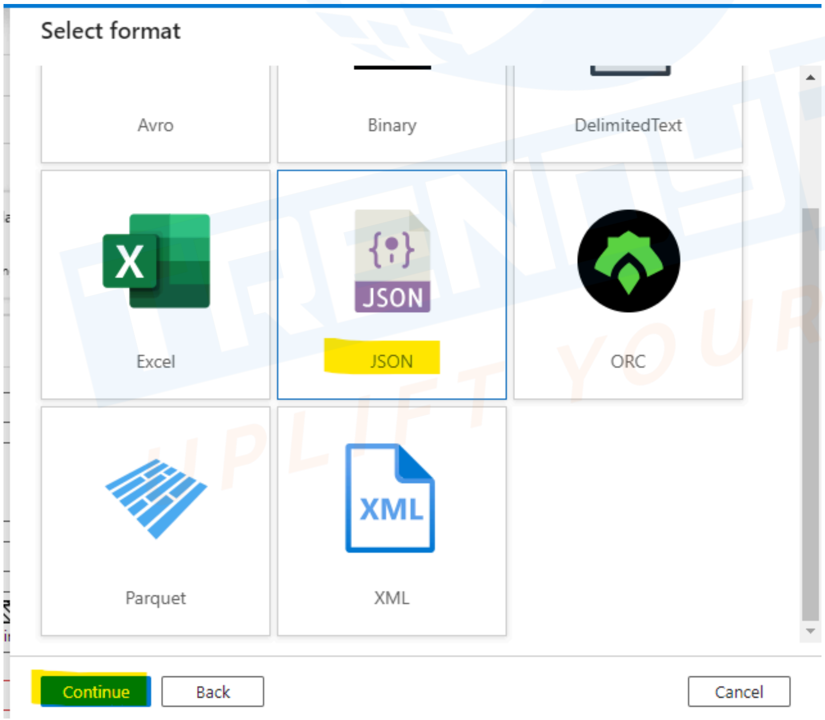
We have to Add the sink but before proceeding ahead we will create the “result” directory in the “data” container in ADLS gen2 storage “trendytechsa” that we have created.

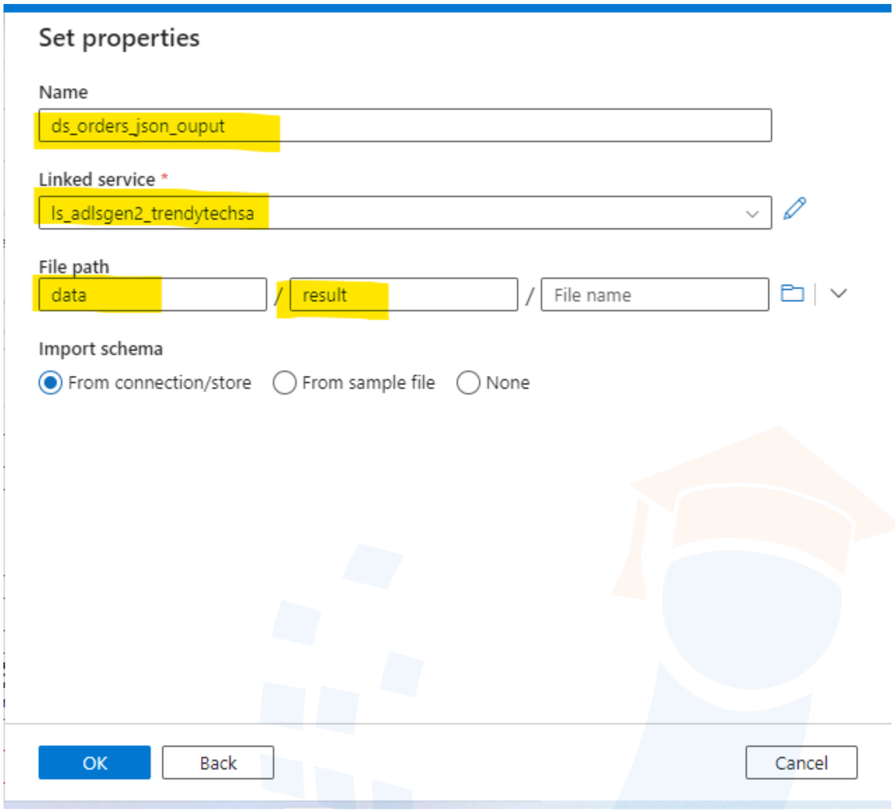


Authentication method: Access key (Switch to Microsoft Entra user account)
Location: data

Name	Modified	Access tier
□ output		
□ result		

Also create dataset **“ds_orders_json_output”** for storing **“orders.json”** data in this **“result“** directory





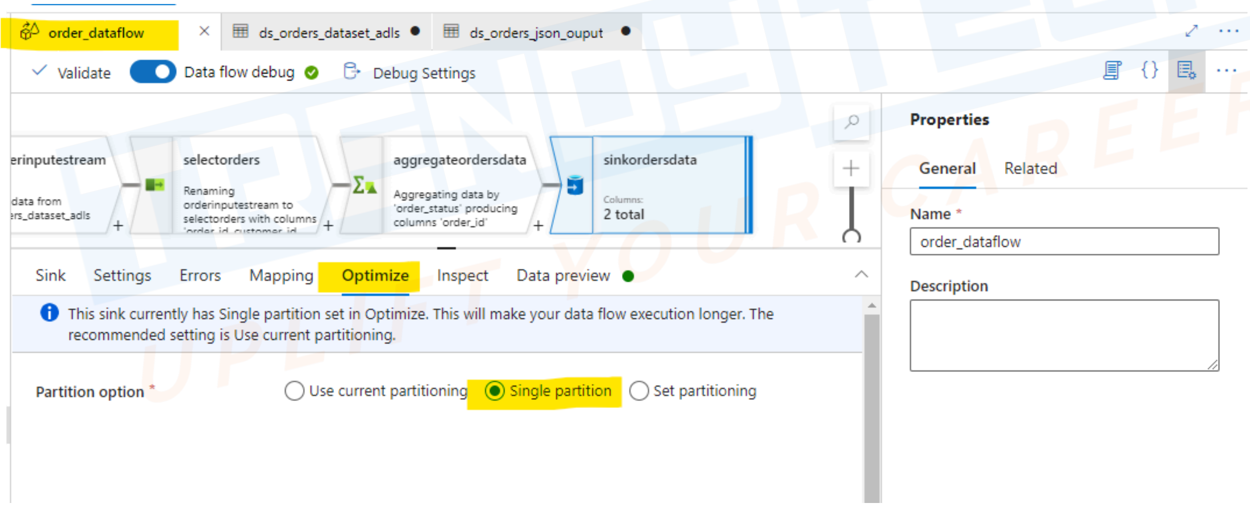
Name: ds_orders_json_ouput

Linked service *: ls_adlsgen2_trendytechsa

File path: data / result / File name

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

For **"orders_dataflow"** click on **"Optimize"** option and select **"Single partitioning"** to store the complete output in a single file.



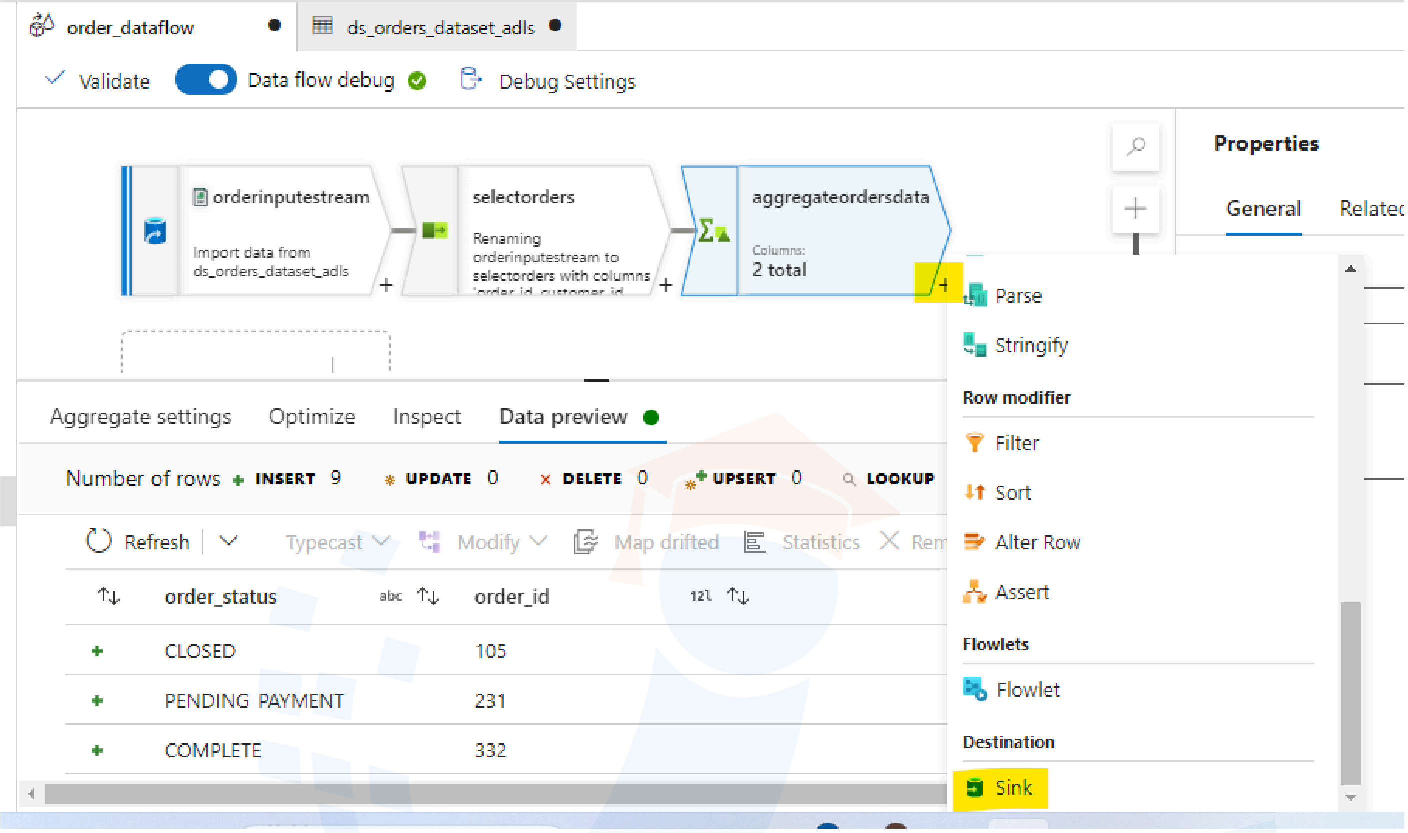
This sink currently has Single partition set in Optimize. This will make your data flow execution longer. The recommended setting is Use current partitioning.

Partition option *: ○ Use current partitioning ● Single partition ○ Set partitioning

Name *: order_dataflow

Description:

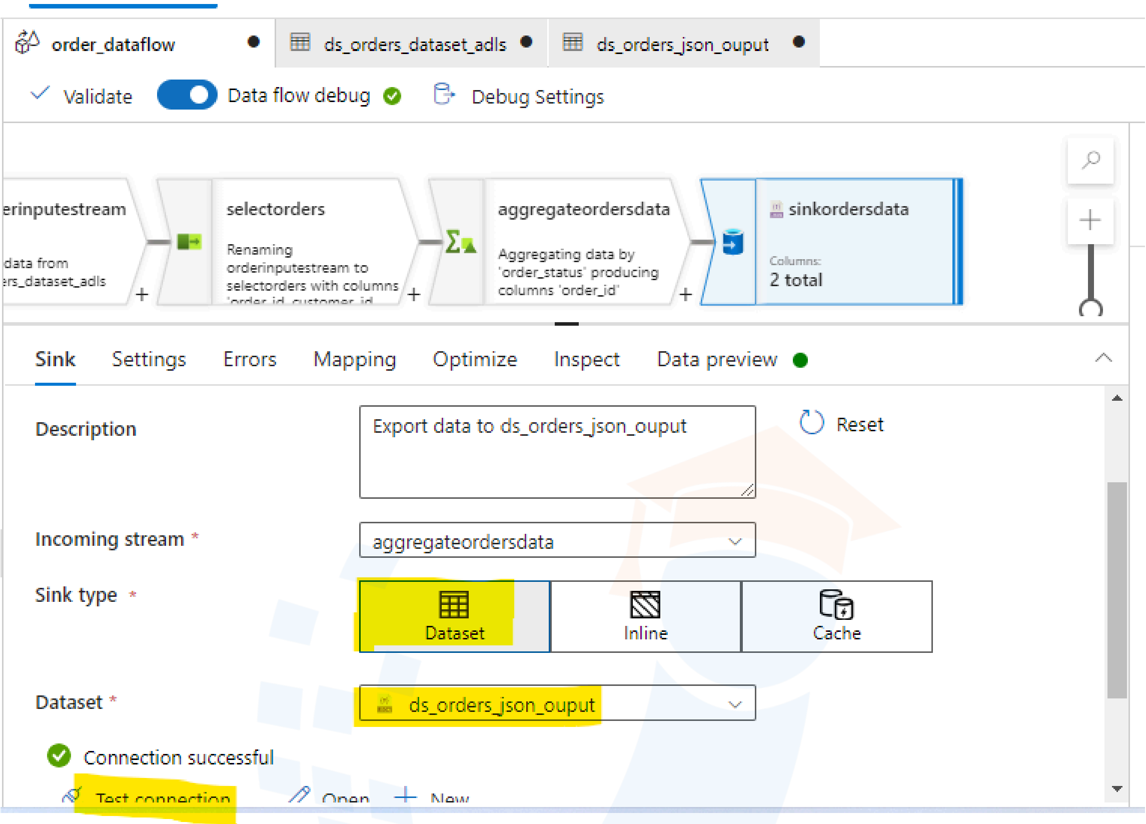
Now click on **"+"** we will add **"sink"** to store the aggregated data refer below screenshot



Number of rows INSERT 9 UPDATE 0 DELETE 0 UPSERT 0 LOOKUP

	order_status	order_id
+	CLOSED	105
+	PENDING PAYMENT	231
+	COMPLETE	332

Add the **“ds_orders_json_output”** in the dataset and test the connection and you can preview the data.



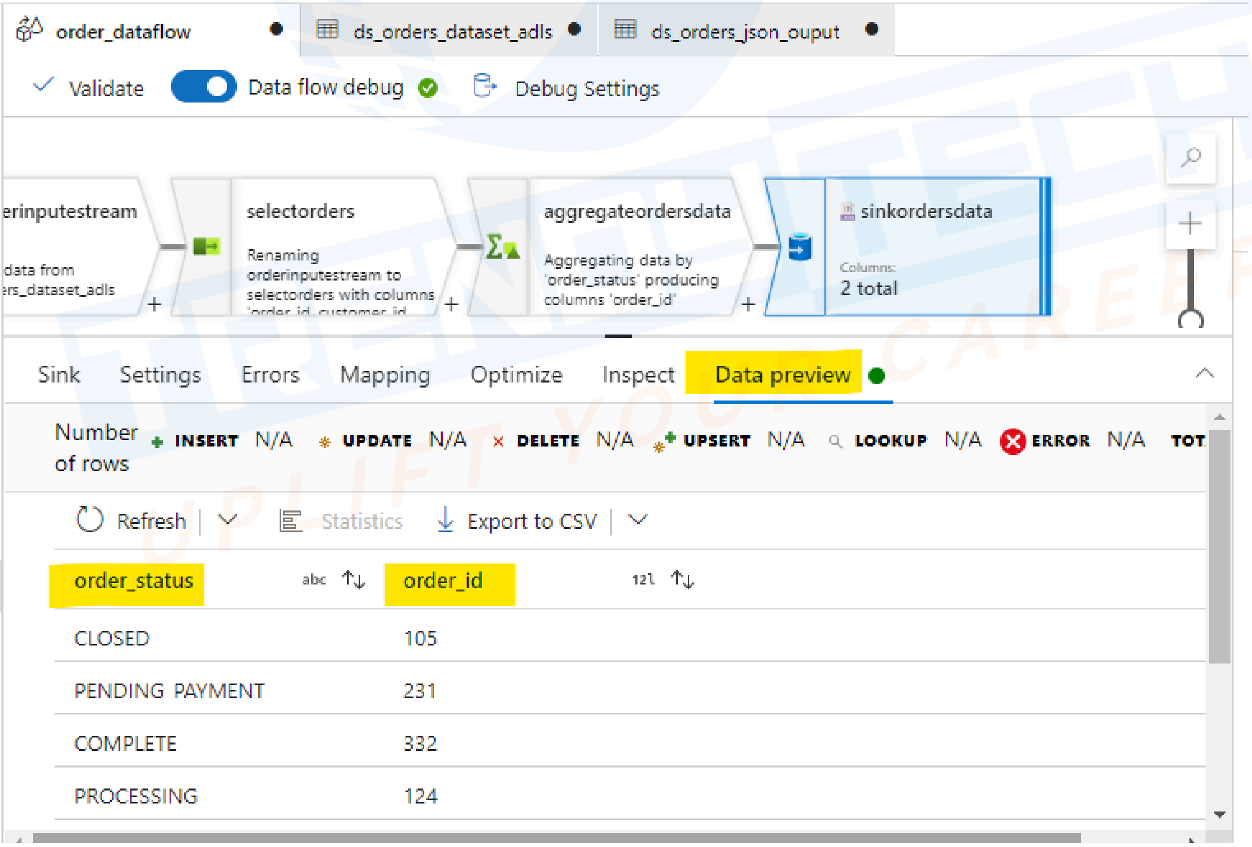
Description: Export data to ds_orders_json_ouput

Incoming stream *: aggregateordersdata

Sink type *: Dataset | Inline | Cache

Dataset *: ds_orders_json_ouput

Connection successful

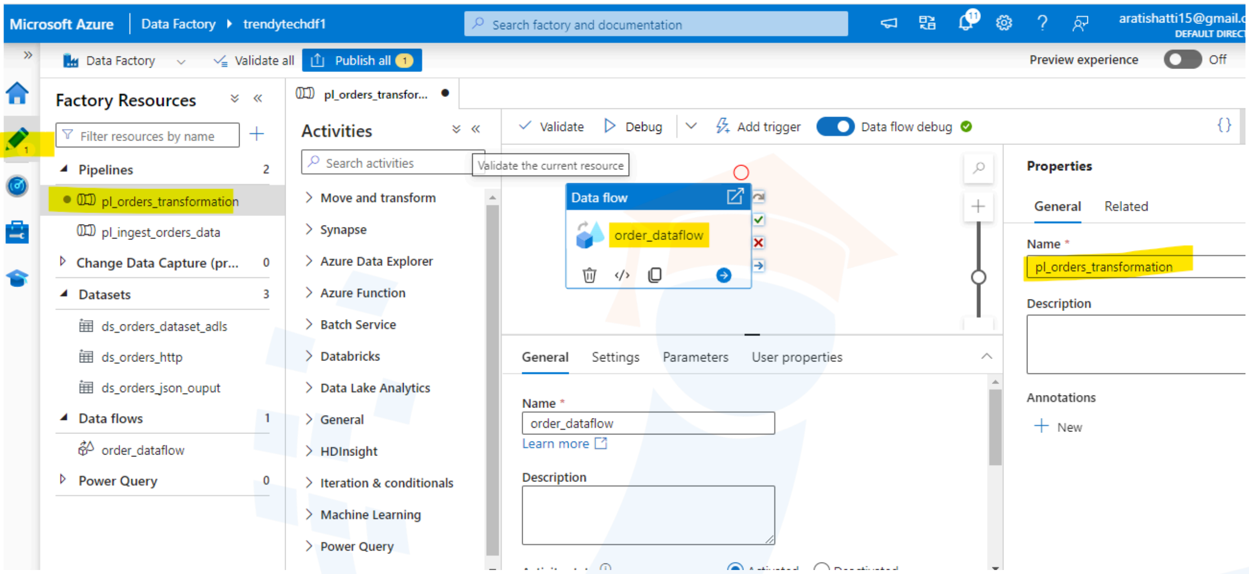


Number of rows INSERT N/A UPDATE N/A DELETE N/A UPSERT N/A LOOKUP N/A ERROR N/A TOT

order_status	order_id
CLOSED	105
PENDING_PAYMENT	231
COMPLETE	332
PROCESSING	124

Note: Before proceeding ahead. be sure to publish these changes to make them active and available for use in your data workflows.

We will create the new pipeline **“pl_orders_transformation”** and will **drag** this dataflow in the pipeline **“pl_orders_transformation”.**

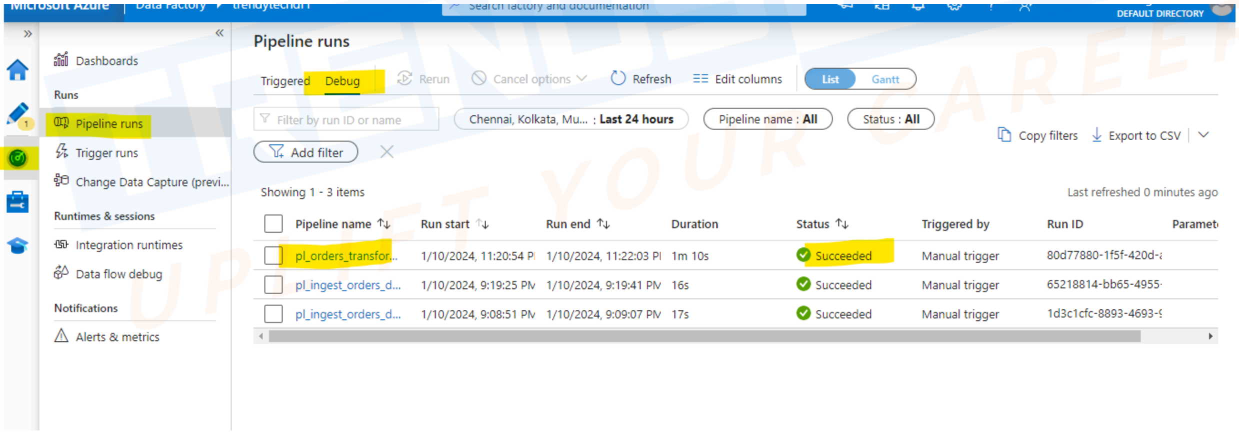


Name * pl_orders_transformation

Name * order_dataflow

Debug the pipeline and validate it.

You can see this running pipeline in the monitor tab (Debug) and publish it.



Pipeline name	Run start	Run end	Duration	Status	Triggered by	Run ID	Paramet
pl_orders_transfor...	1/10/2024, 11:20:54 PI	1/10/2024, 11:22:03 PI	1m 10s	Succeeded	Manual trigger	80d77880-1f5f-420d-ɛ	
pl_ingest_orders_d...	1/10/2024, 9:19:25 PM	1/10/2024, 9:19:41 PM	16s	Succeeded	Manual trigger	65218814-bb65-4955-	
pl_ingest_orders_d...	1/10/2024, 9:08:51 PM	1/10/2024, 9:09:07 PM	17s	Succeeded	Manual trigger	1d3c1cfc-8893-4693-ʢ	

Note: After creating pipelines in Azure Data Factory, be sure to publish these changes to make them active and available for use in your data workflows.