PROJECT

Data Integration & Transformation

- we have 3 tables customer, product, and sales. Here I choose retail domain you can take any domain like health care, and financial data etc.
- Customer.sql table in azure SQL database, product.json table in blob storage, and sales.dbo table in an on-premises SQL server database.
- Here I choose data lake is a unified storage system. So we have to ingest data from these sources to a data lake and transform different data file formats to parquet file format.
- So we can easily analyze the data and generate insights from data.
- The tech stack I have used for this azure data factory, azure data lake, azure SQL database, SQL server.

Azure data factory :- Azure Data Factory is a cloud-based data integration and ETL (Extract, Transform, Load) service by Microsoft for orchestration and automation of data integration and transformation.

Azure data lake:- Azure Data Lake is a cloud-based big data storage and analytics service by Microsoft that allows storing, processing, and analyzing large amounts of structured, semi-structured, and unstructured data.

Azure SQL database:- Azure SQL Database is a fully managed relational database service by Microsoft that provides a scalable and secure platform for storing, processing, and managing structured data, based on the SQL Server engine.

Azure blob storage:- Azure Blob Storage is a cloud-based object storage service by Microsoft for unstructured data such as text and binary data, images, audio and video files, which can be accessed via HTTP/HTTPS from anywhere in the world.

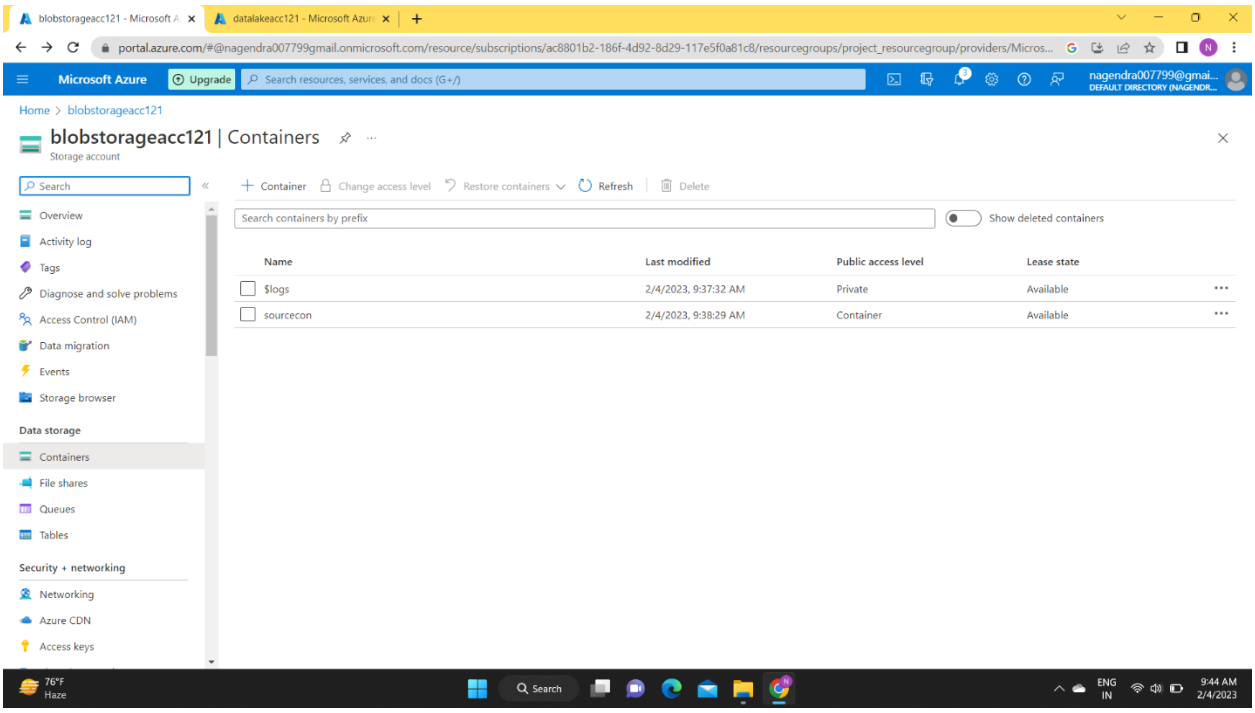
Data integration:- Data integration is the process of combining data from various sources into a unified, centralized repository for analysis and reporting.

Data transformation:- Data transformation is the process of converting data from one format or structure to another to make it usable for analysis or reporting.

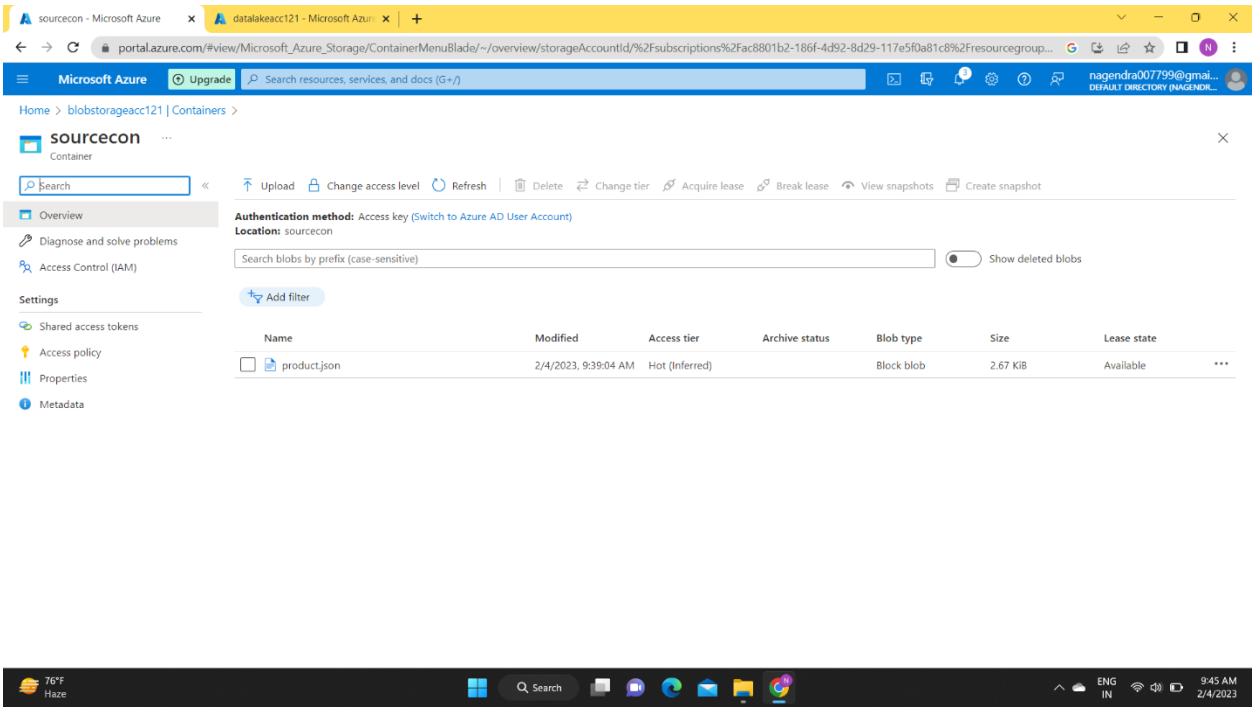
Linked service:- A linked service in Azure Data Factory is a connection entity that defines the relationship between the data factory and an external data store, such as a database, file system, or cloud storage. It contains the connection details and authentication information needed to access the external data store.

Data set:- A dataset in Azure Data Factory represents the structure and metadata of data stored in a data store, such as a database table or file system, and is used as the basis for defining data transformations and movement activities in data pipelines.

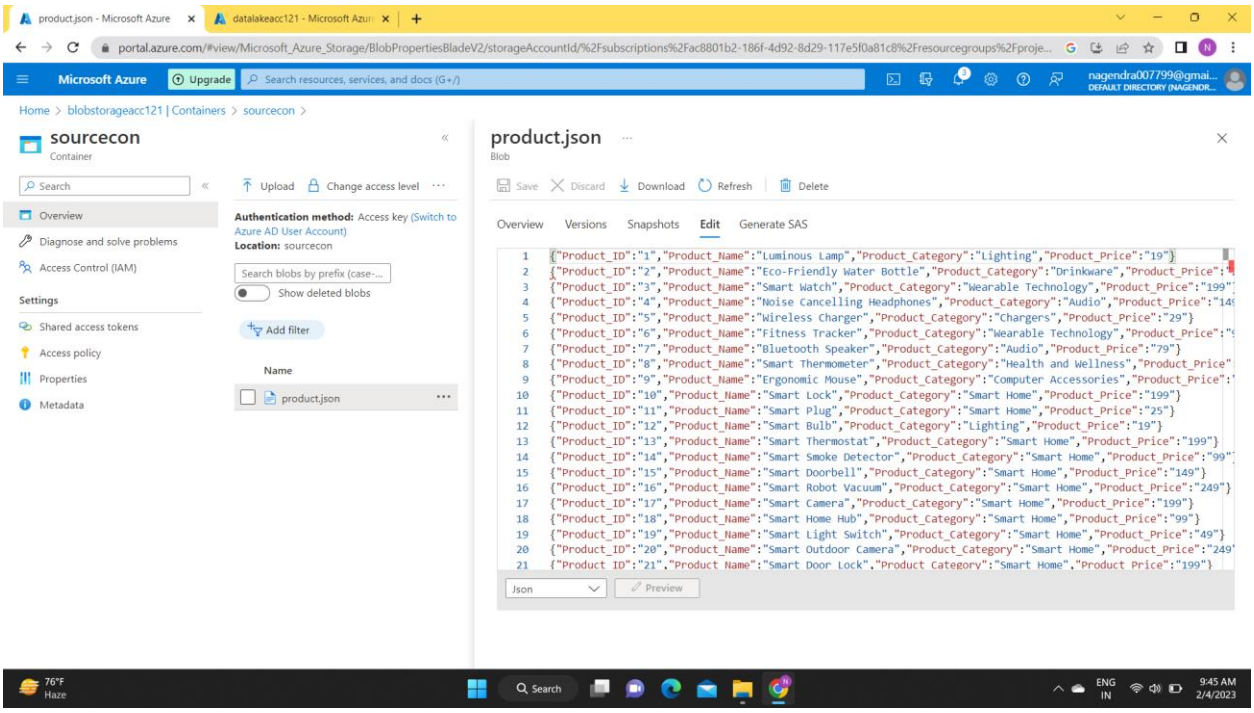
- This is my blob storage “blobstorageacc121” and container name ‘sourcecon’.



- In this blob storage, sourcecon we have product table. The product table is in json file format.

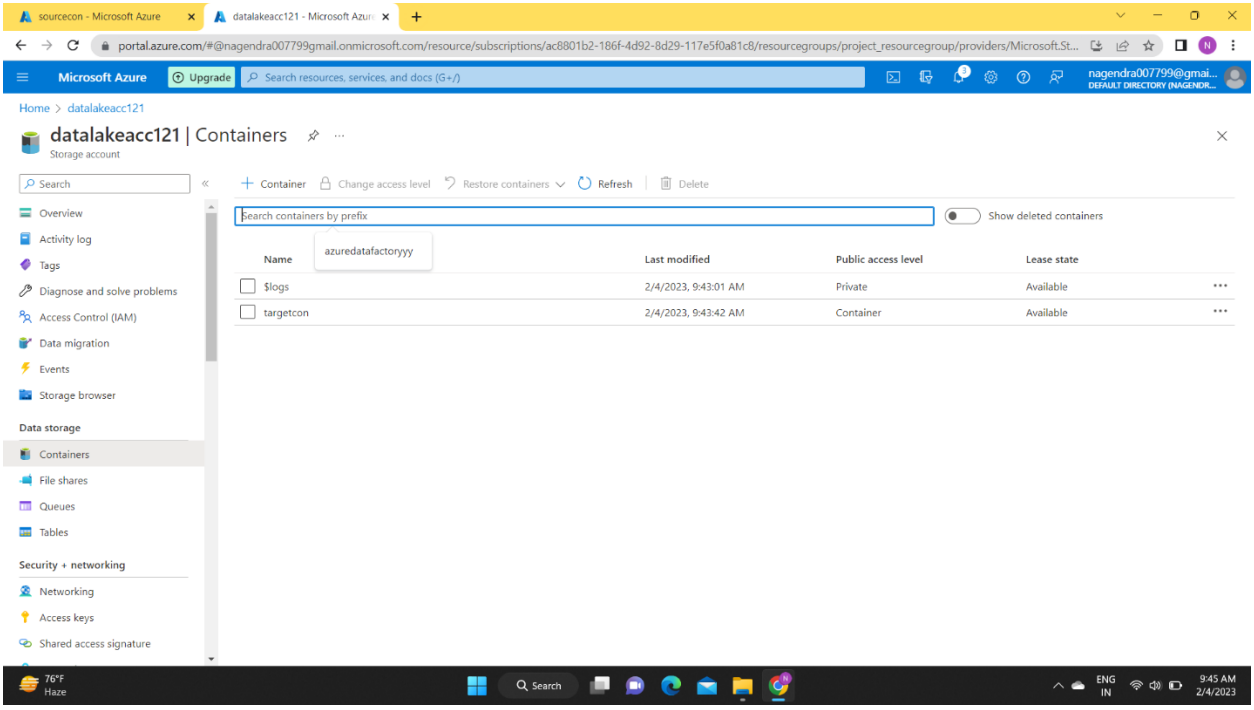


- Here we can see the data in the product table

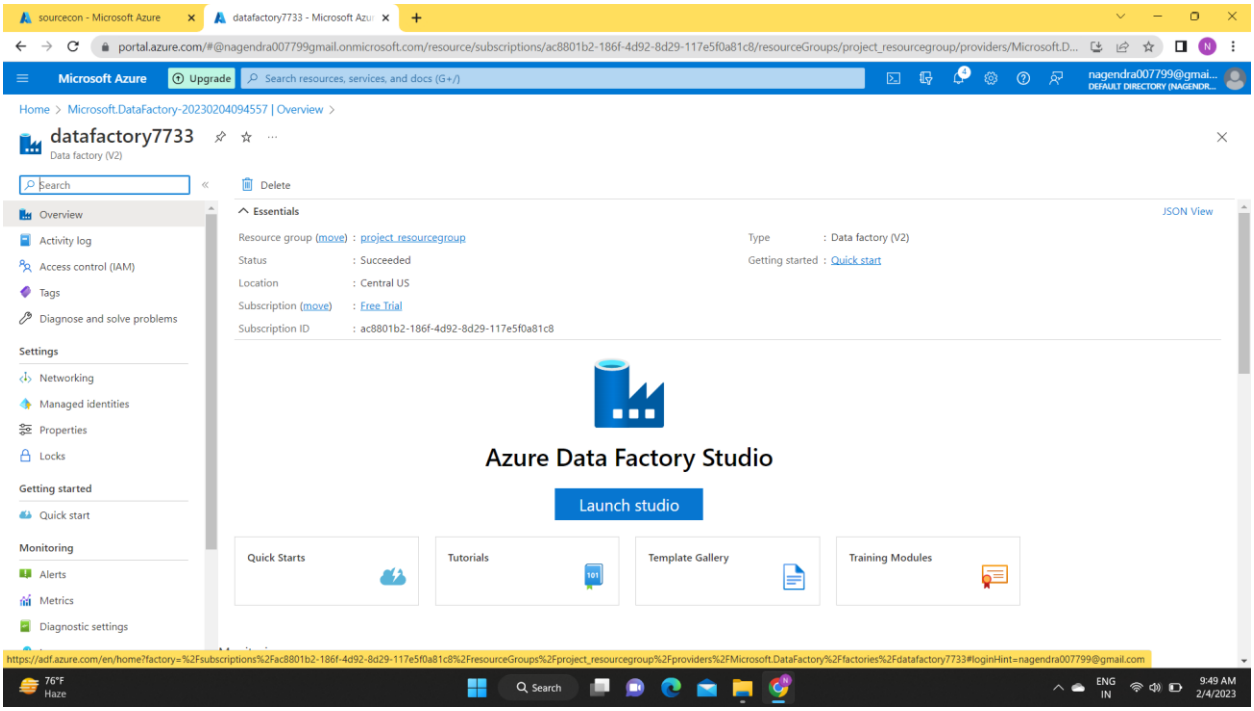


- Here data lake is our unified storage account

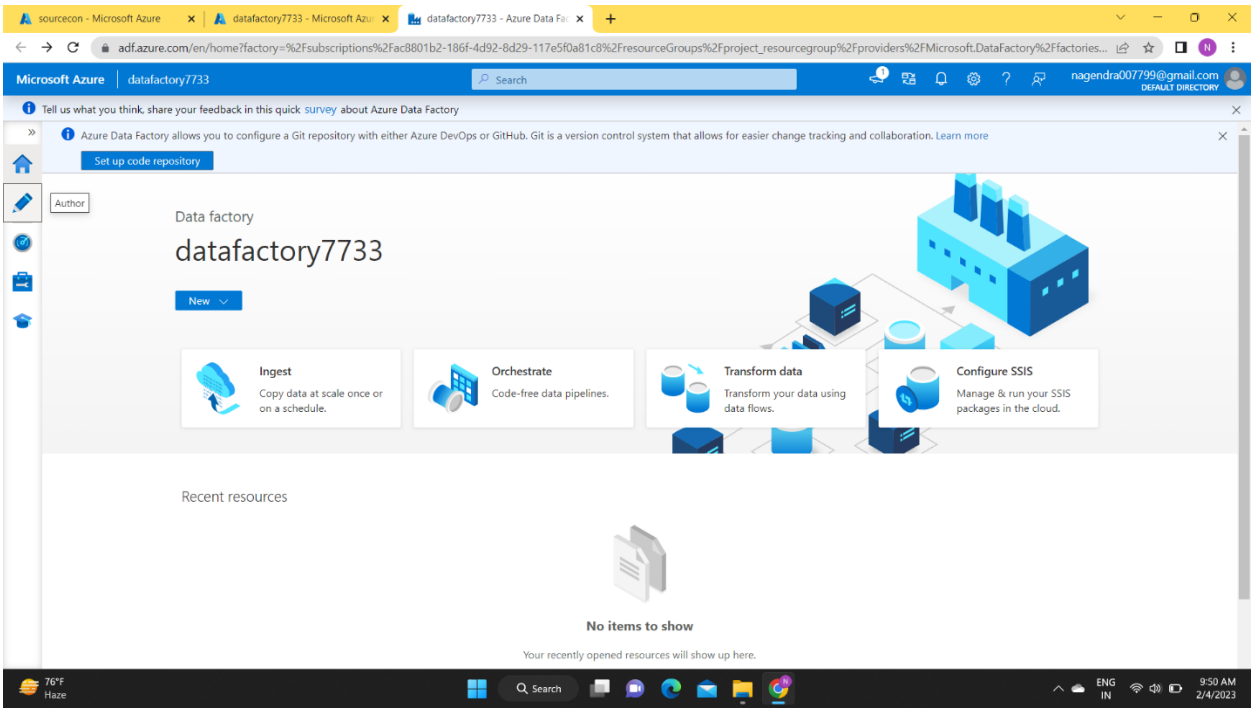
- Data lake name is “datalakeacc121” and container name is ‘targetcon’



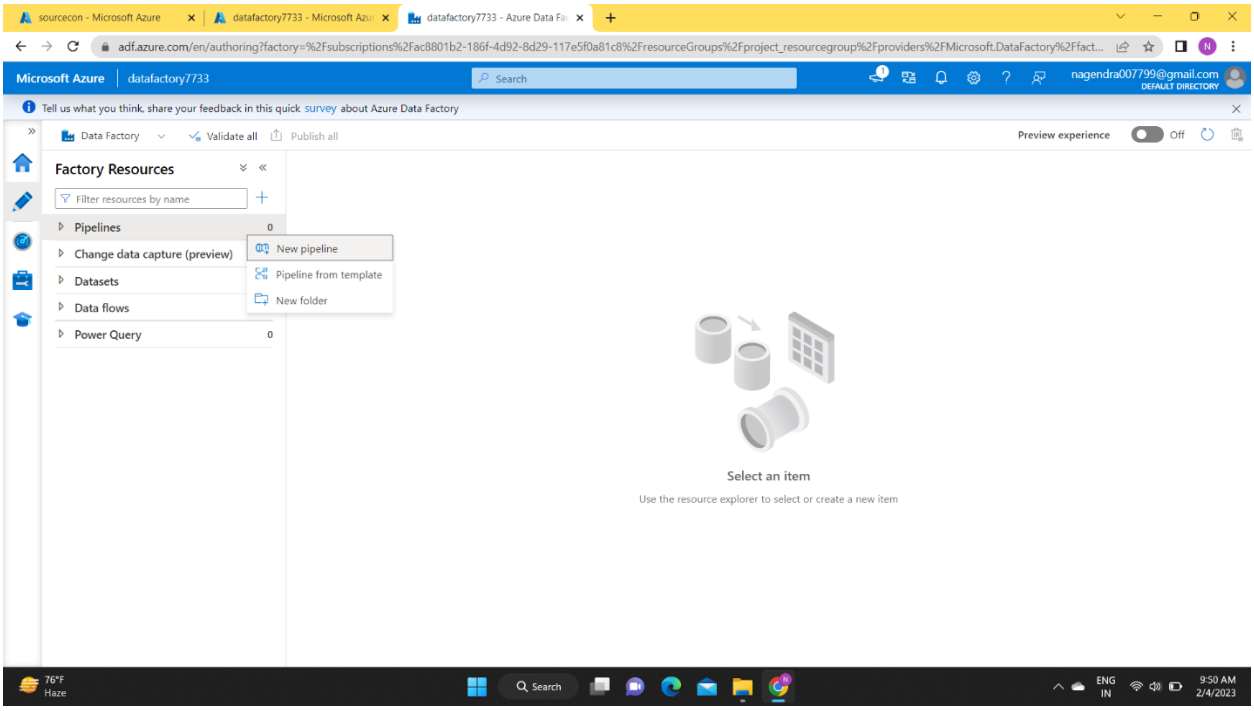
- Our azure data factory name is “datafactory7733”
- We are launching our datafactory by clicking “launch studio”



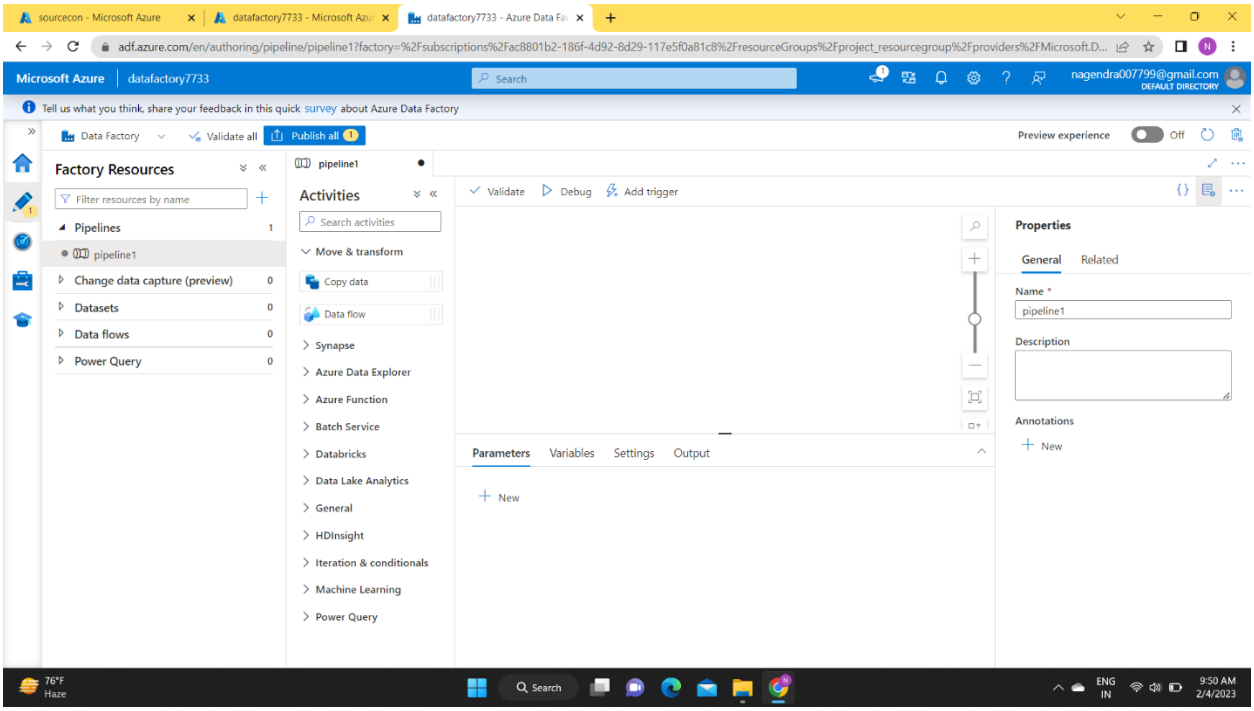
- Go to author tab



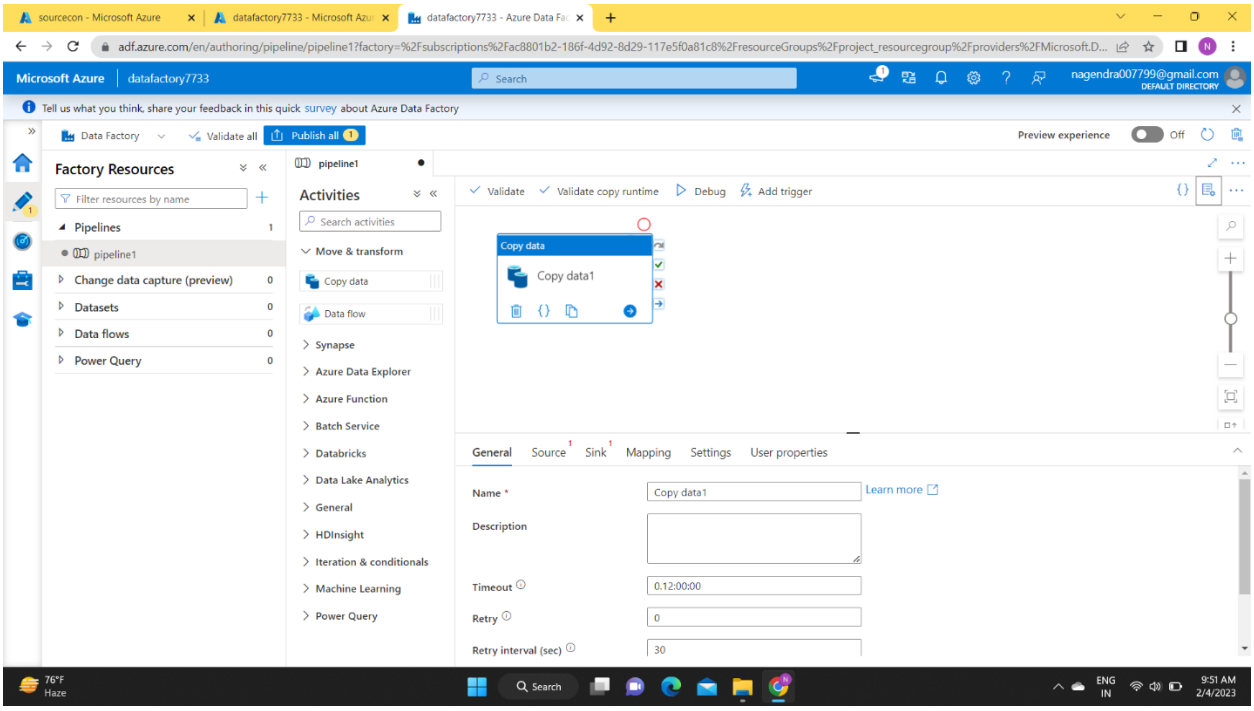
- In author go to pipeline and select new pipeline



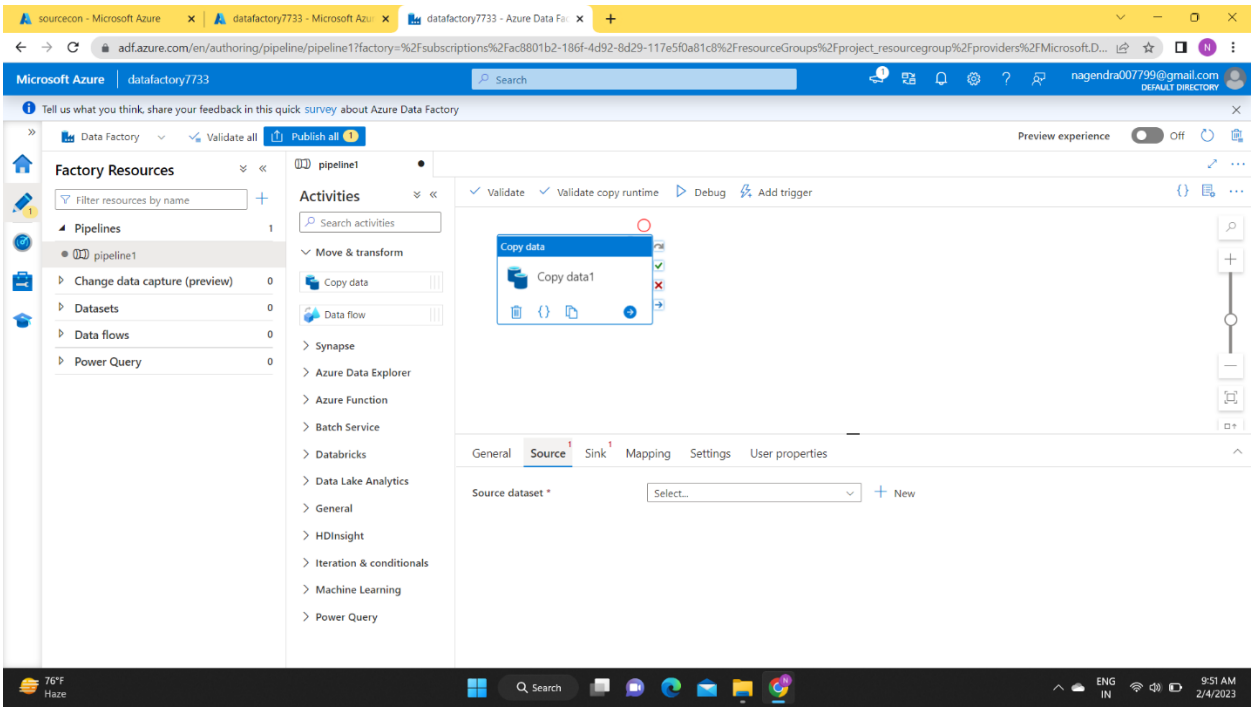
- Launch new pipeline
- Default pipeline name as “pipeline1”



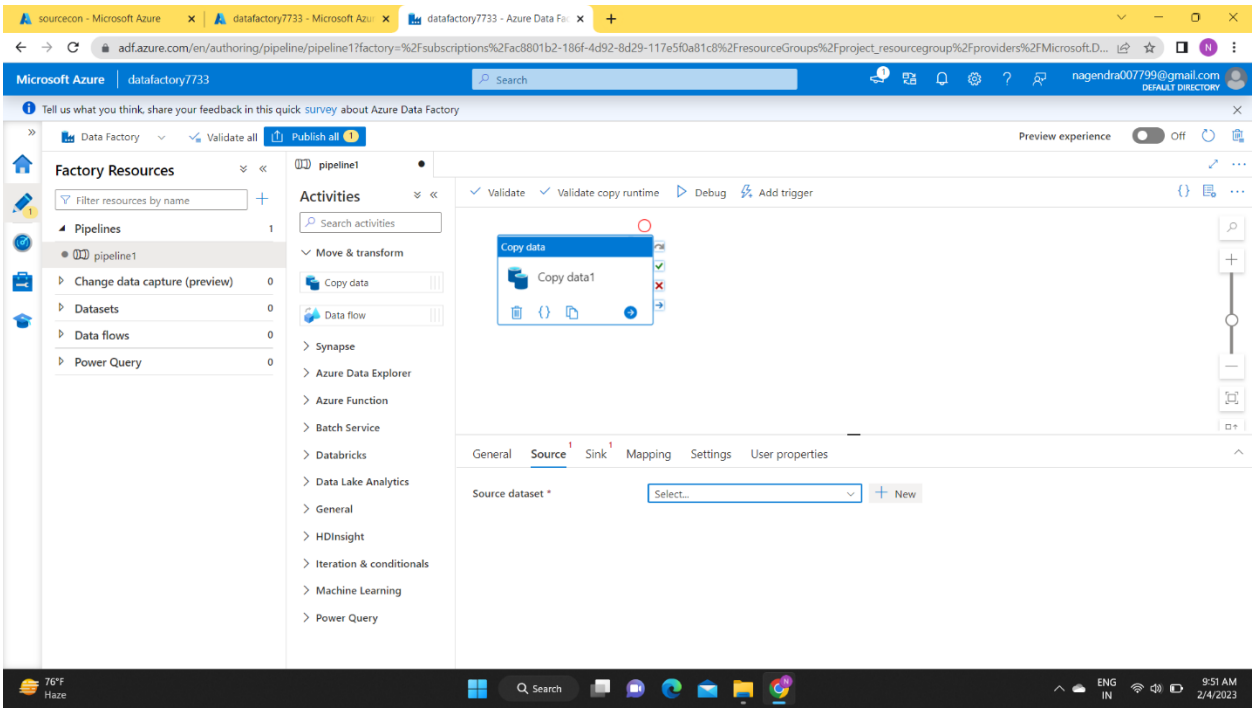
- In move and transform select copy data activity



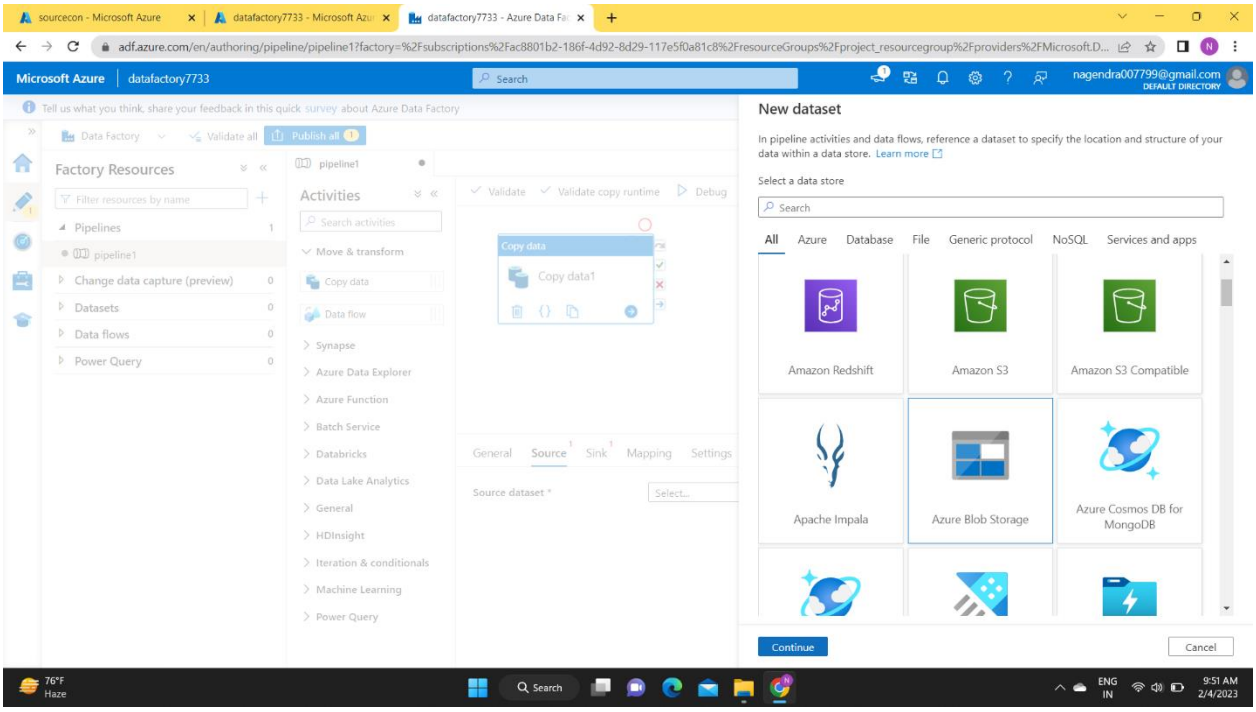
- Click on copy data and select source



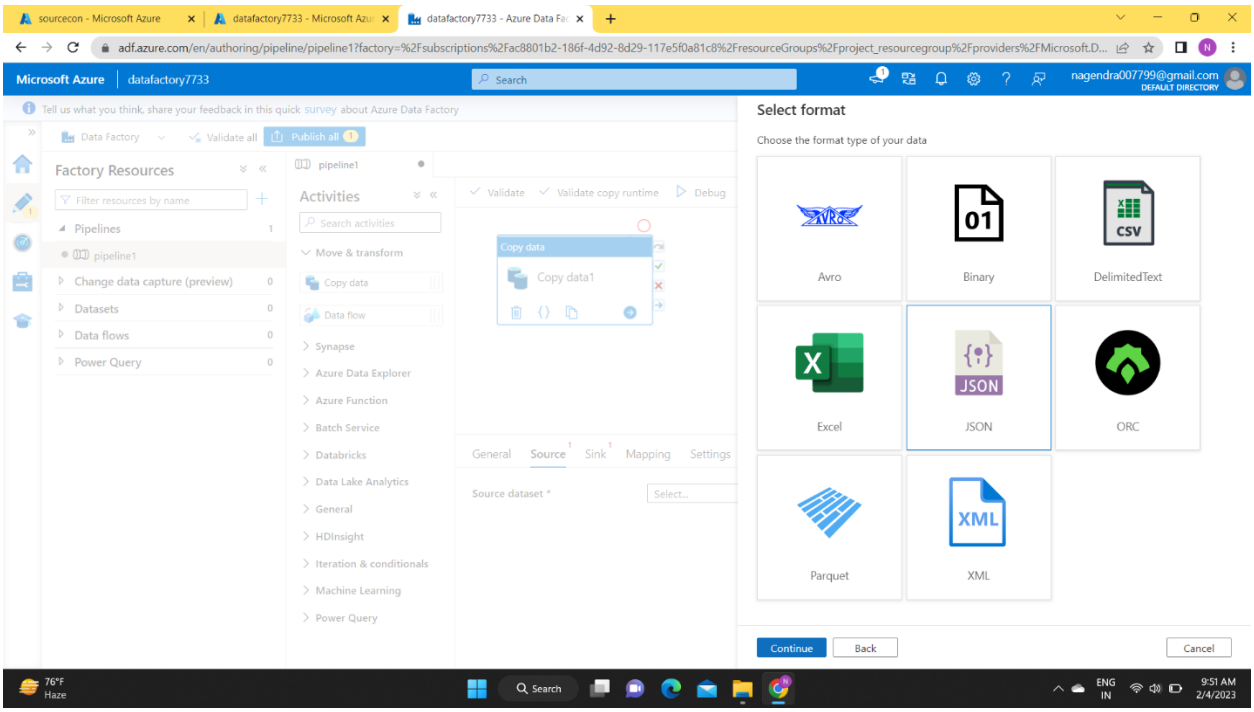
- In dataset box select +new



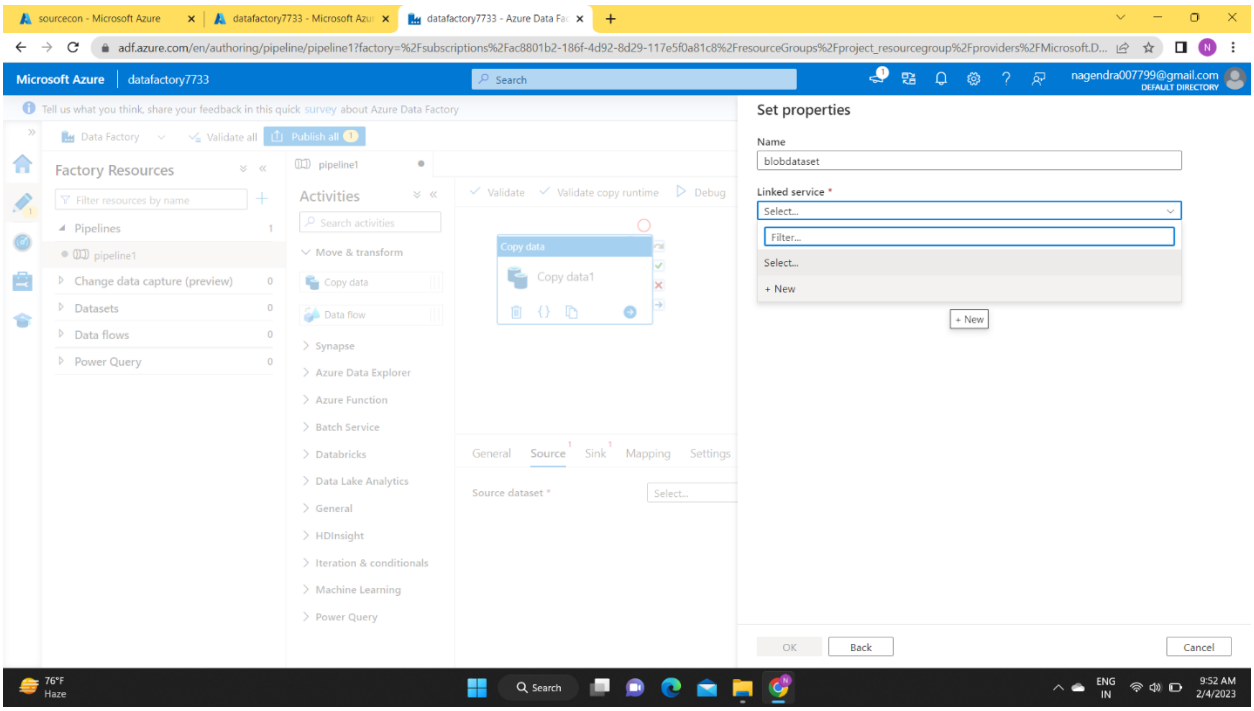
- Select the source storage type
- Here our source is blob storage and press continue.



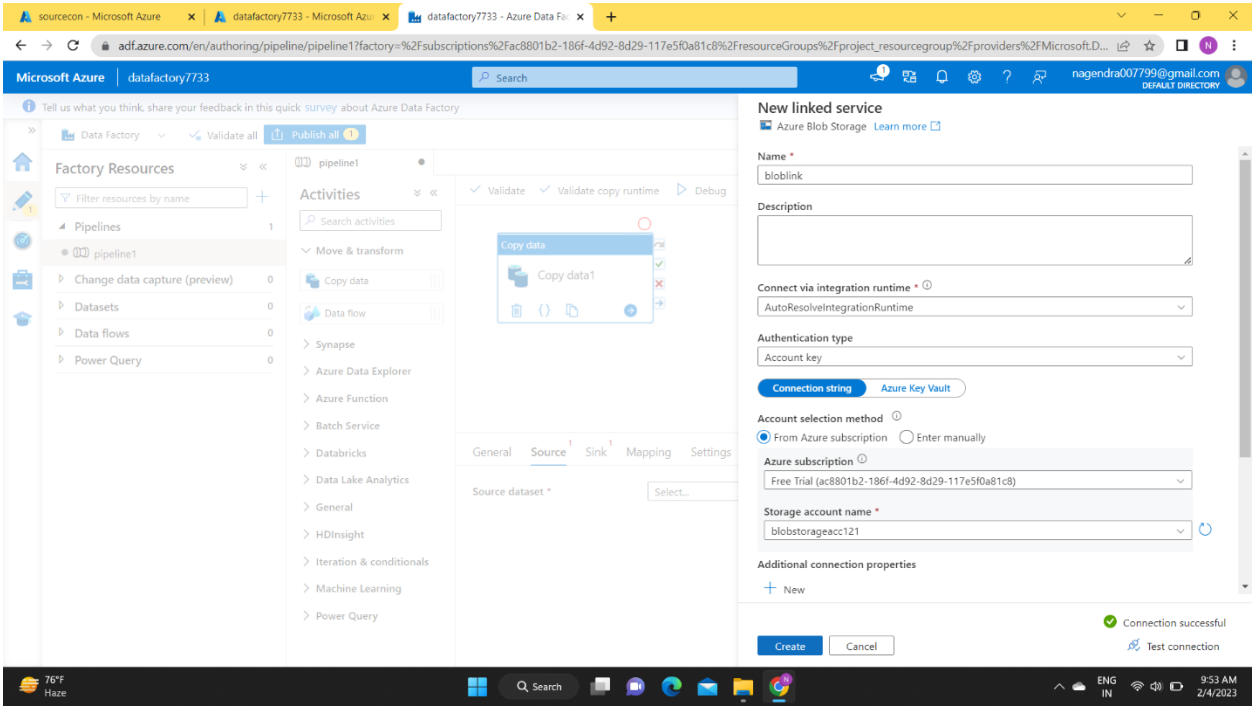
- After the source storage type we have to choose the source file format type
- Our source file format type is json.



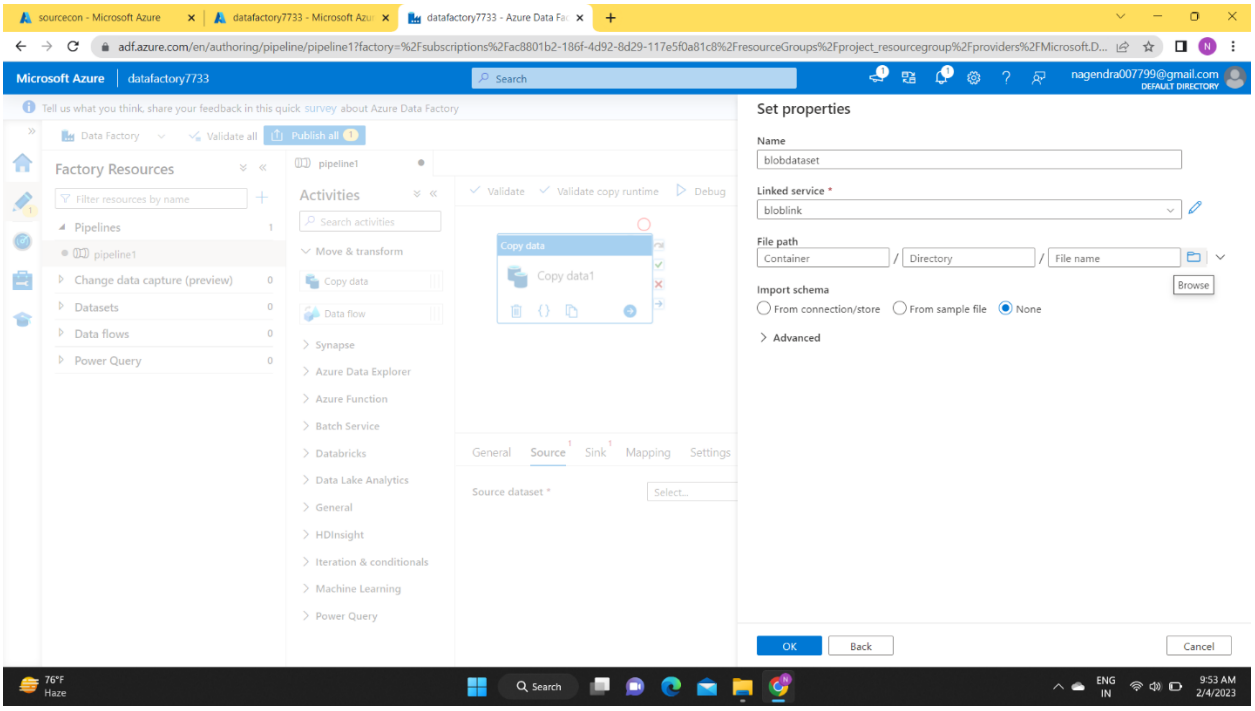
- In data set we have to give link service
- So we are creating new link service by clicking +new

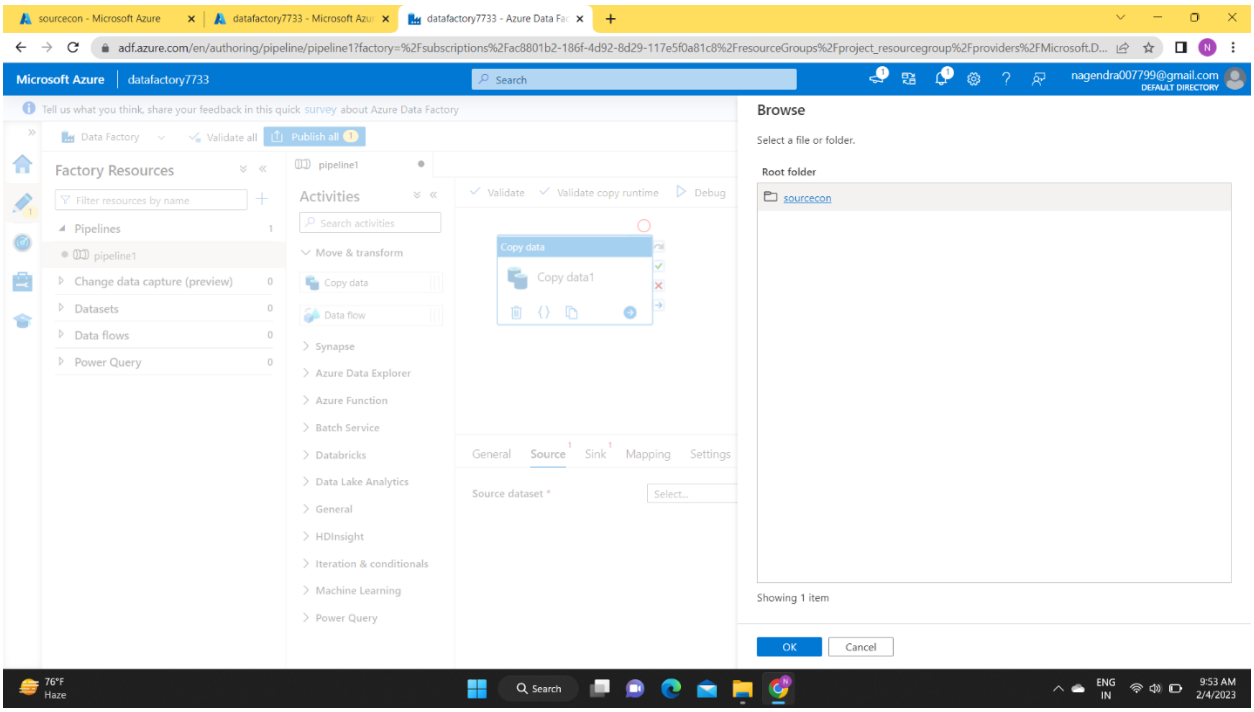


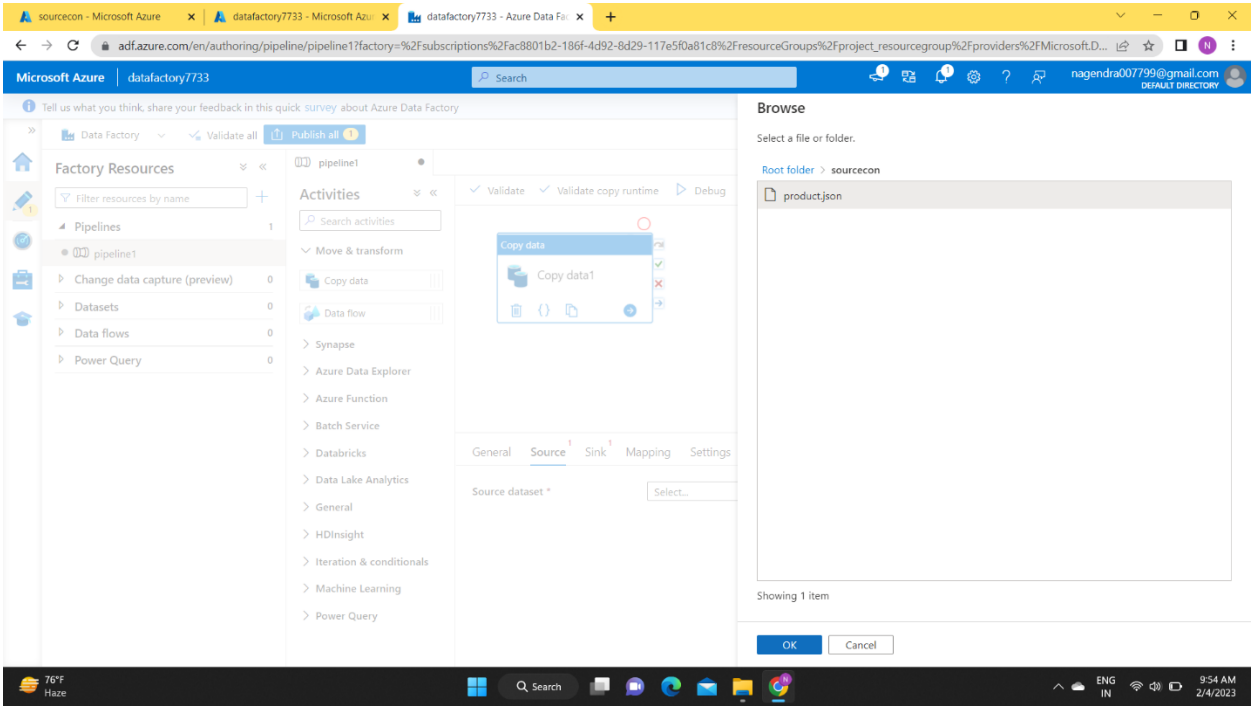
- Here you give name of your link service “bloblink”, and give azure subscription and storage account name.
- Must do the test connection.



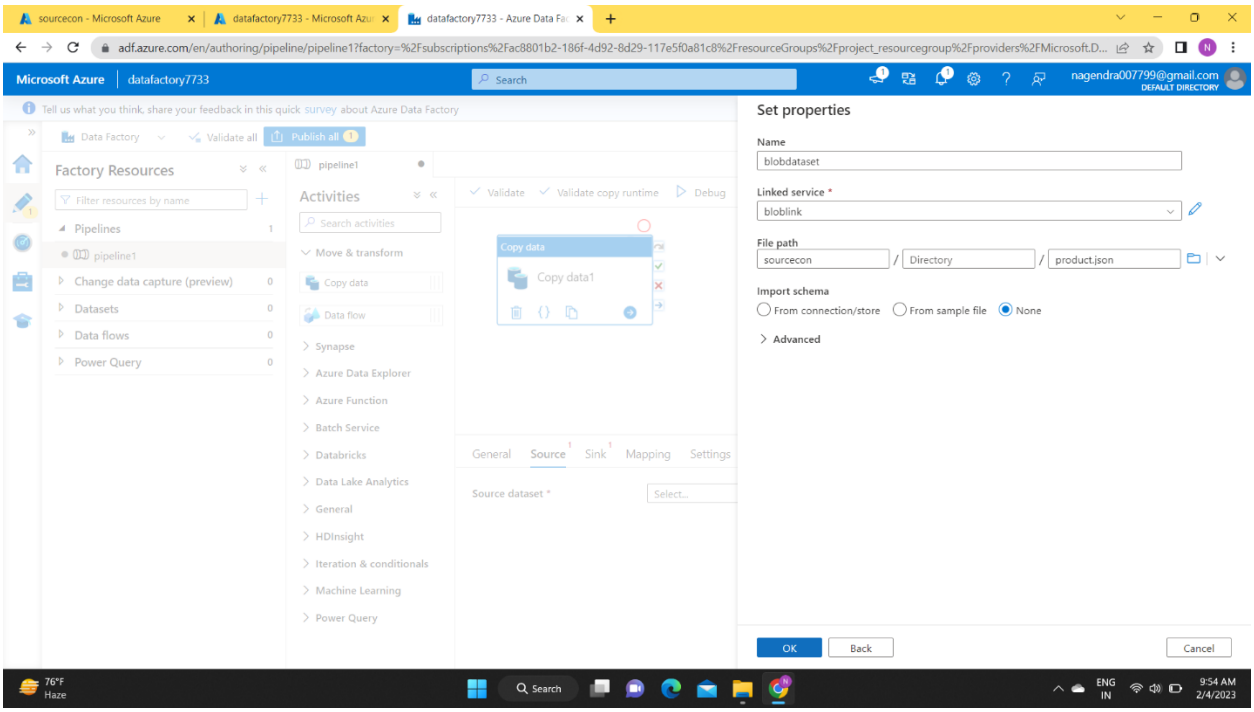
- After that select the source file location.



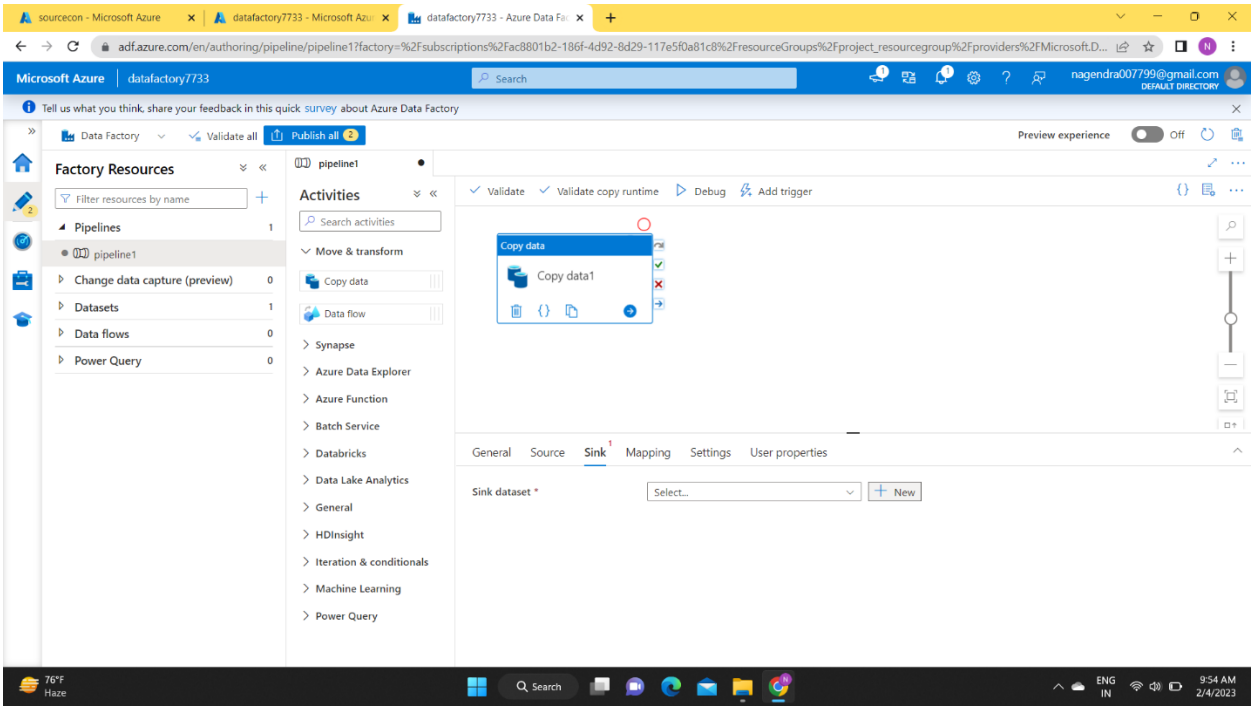

sourcecon - Microsoft Azure
datafactory7733 - Microsoft Azu
datafactory7733 - Azure Data Fa
adf.azure.com/en/authoring/pipeline/pipeline1?factory=%2Fsubscriptions%2Fac8801b2-186f-4d92-8d29-117e5f0a81c8%2FresourceGroups%2Fproject_resourcegroup%2Fproviders%2FMicrosoft.D...
Microsoft Azure
datafactory7733
Search
nagendra007799@gmail.com
DEFAULT DIRECTORY
Tell us what you think, share your feedback in this quick survey about Azure Data Factory
Data Factory
Validate all
Publish all
Factory Resources
Filter resources by name
Pipelines
pipeline1
Change data capture (preview)
Datasets
Data flows
Power Query
Activities
Search activities
Move & transform
Copy data
Data flow
Synapse
Azure Data Explorer
Azure Function
Batch Service
Databricks
Data Lake Analytics
General
HDInsight
Iteration & conditionals
Machine Learning
Power Query
Validate
Validate copy runtime
Debug
Copy data1
General
Source
Sink
Mapping
Settings
Source dataset *
Select...
Browse
Select a file or folder.
Root folder
sourcecon
Showing 1 item
OK
Cancel
76°F
Haze
Search
ENG
IN
9:53 AM
2/4/2023


sourcecon - Microsoft Azure
datafactory7733 - Microsoft Azu
datafactory7733 - Azure Data Fa
adf.azure.com/en/authoring/pipeline/pipeline1?factory=%2Fsubscriptions%2Fac8801b2-186f-4d92-8d29-117e5f0a81c8%2FresourceGroups%2Fproject_resourcegroup%2Fproviders%2FMicrosoft.D...
Microsoft Azure
datafactory7733
Search
nagendra007799@gmail.com
DEFAULT DIRECTORY
Tell us what you think, share your feedback in this quick survey about Azure Data Factory
Data Factory
Validate all
Publish all
Factory Resources
Filter resources by name
Pipelines
pipeline1
Change data capture (preview)
Datasets
Data flows
Power Query
Activities
Search activities
Move & transform
Copy data
Data flow
Synapse
Azure Data Explorer
Azure Function
Batch Service
Databricks
Data Lake Analytics
General
HDInsight
Iteration & conditionals
Machine Learning
Power Query
Validate
Validate copy runtime
Debug
Copy data1
General
Source
Sink
Mapping
Settings
Source dataset *
Select...
Browse
Select a file or folder.
Root folder > sourcecon
product.json
Showing 1 item
OK
Cancel
76°F
Haze
Search
ENG
IN
9:54 AM
2/4/2023

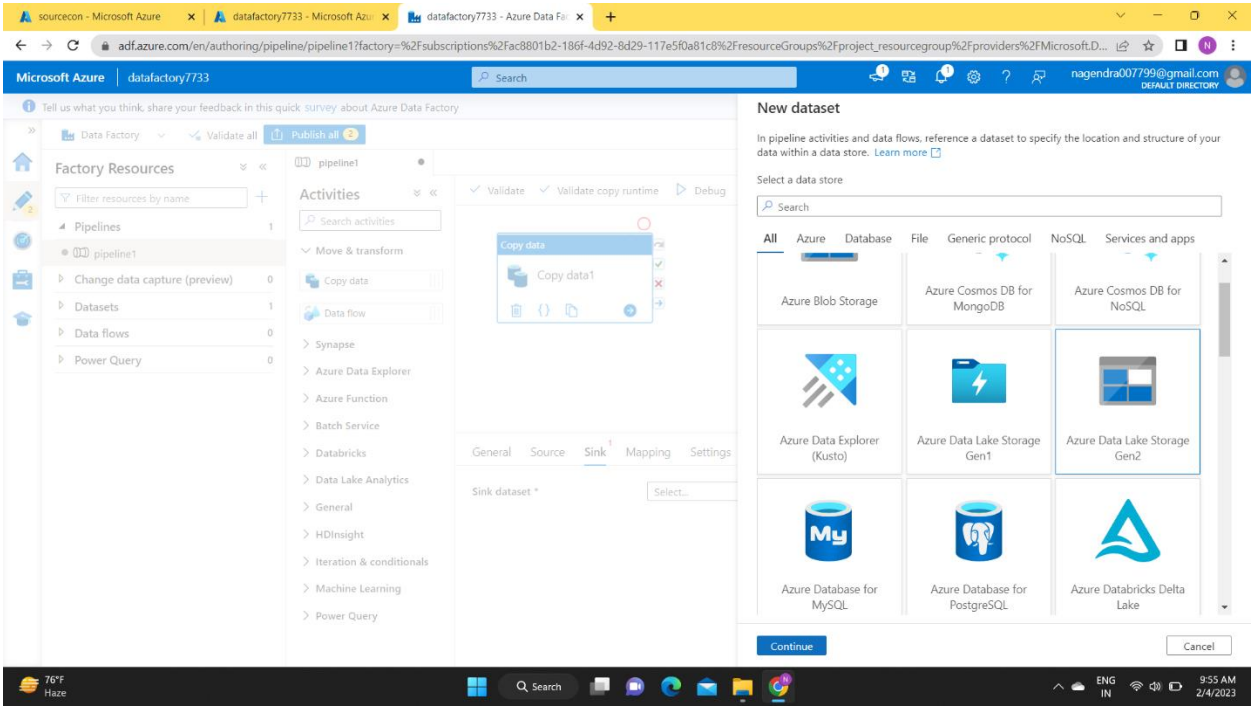
- Don’t select “from connection” in “import schema” put it “none”
- Because we are transferring data from “json to parquet” file format.



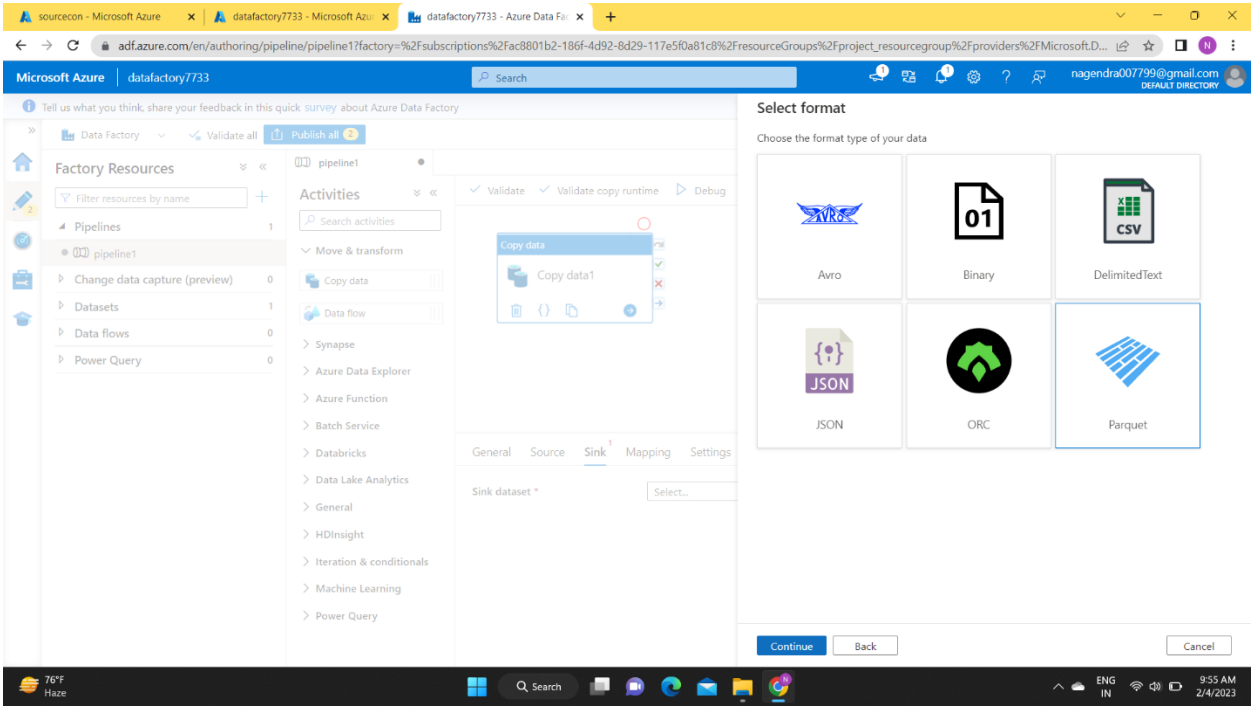
- Now configure sink (target or unified storage)



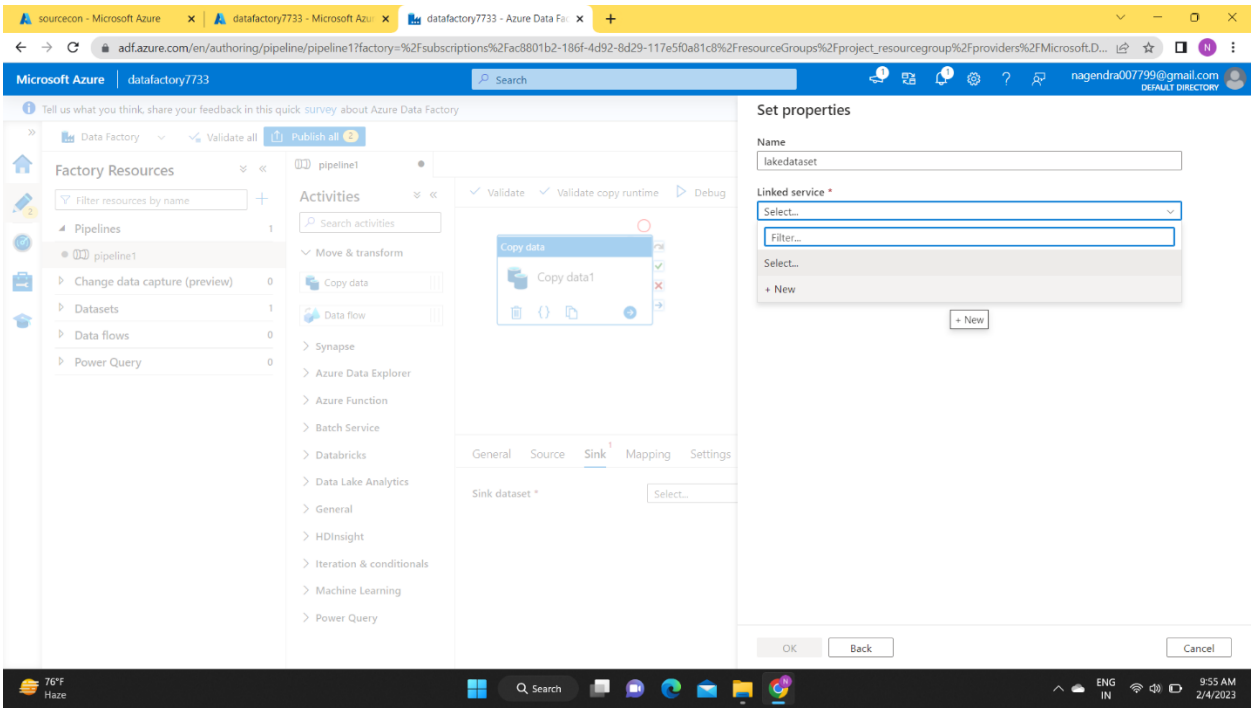
- Our target is datalake,so select datalake.



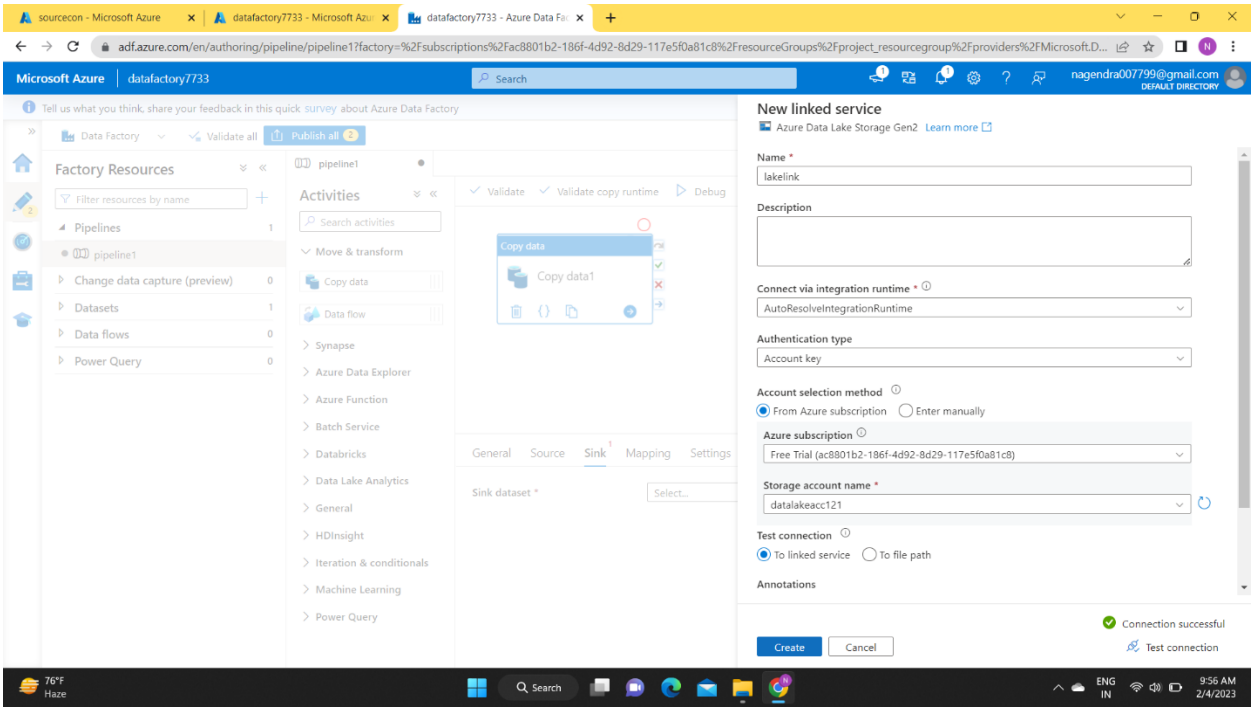
- Our final file format is parquet, so select parquet.



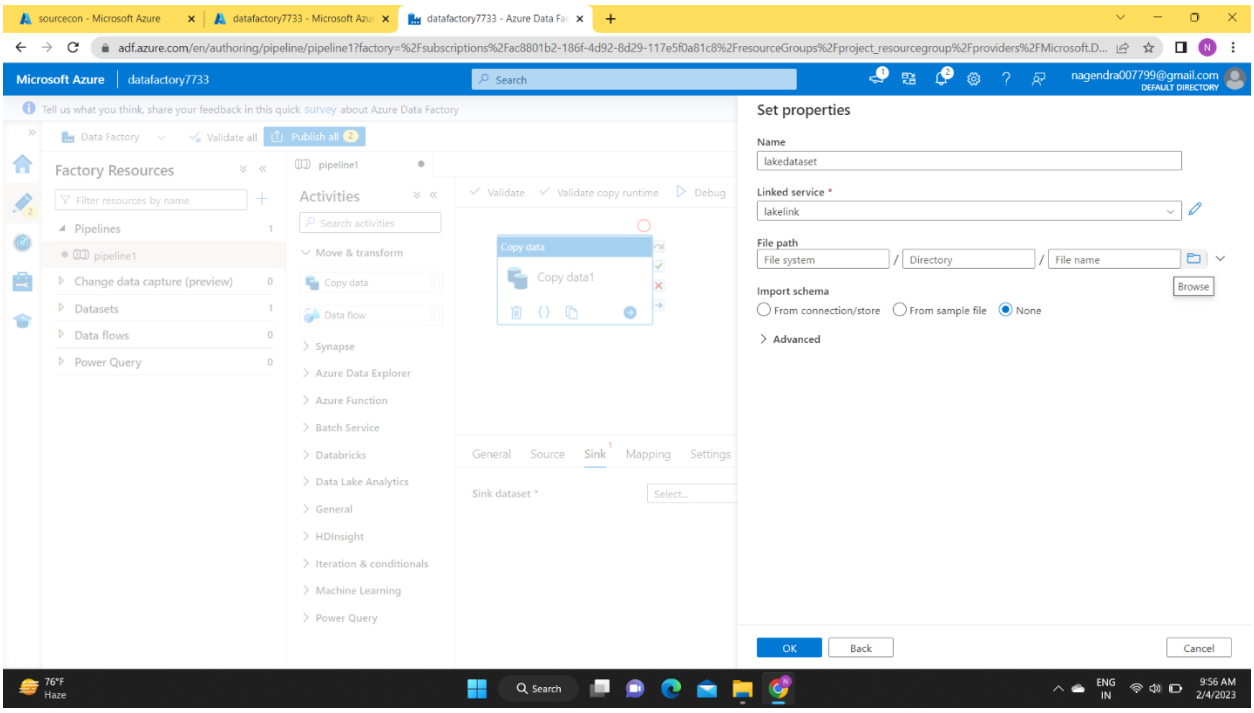
- Our dataset name is “lakedataset” and create a new link service for our target datalake.

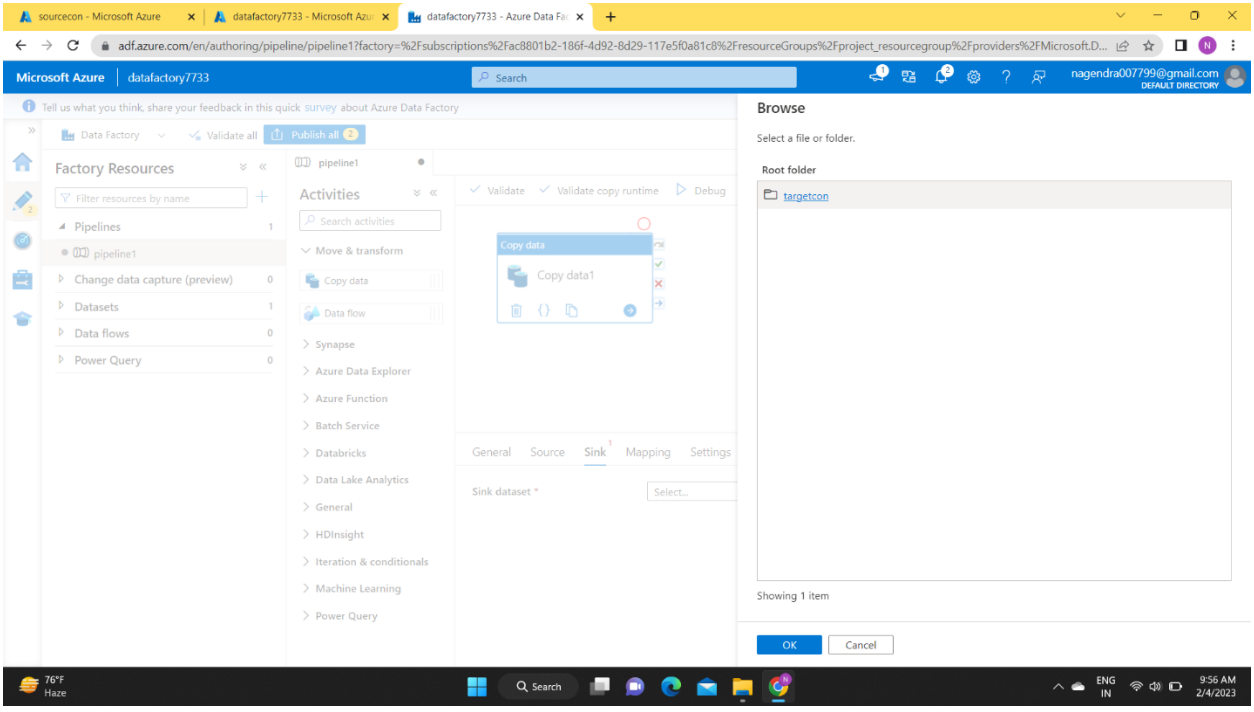


- Our target(data lake) link service name is “lakelink”, also select the azure subscription and storage account of our target.

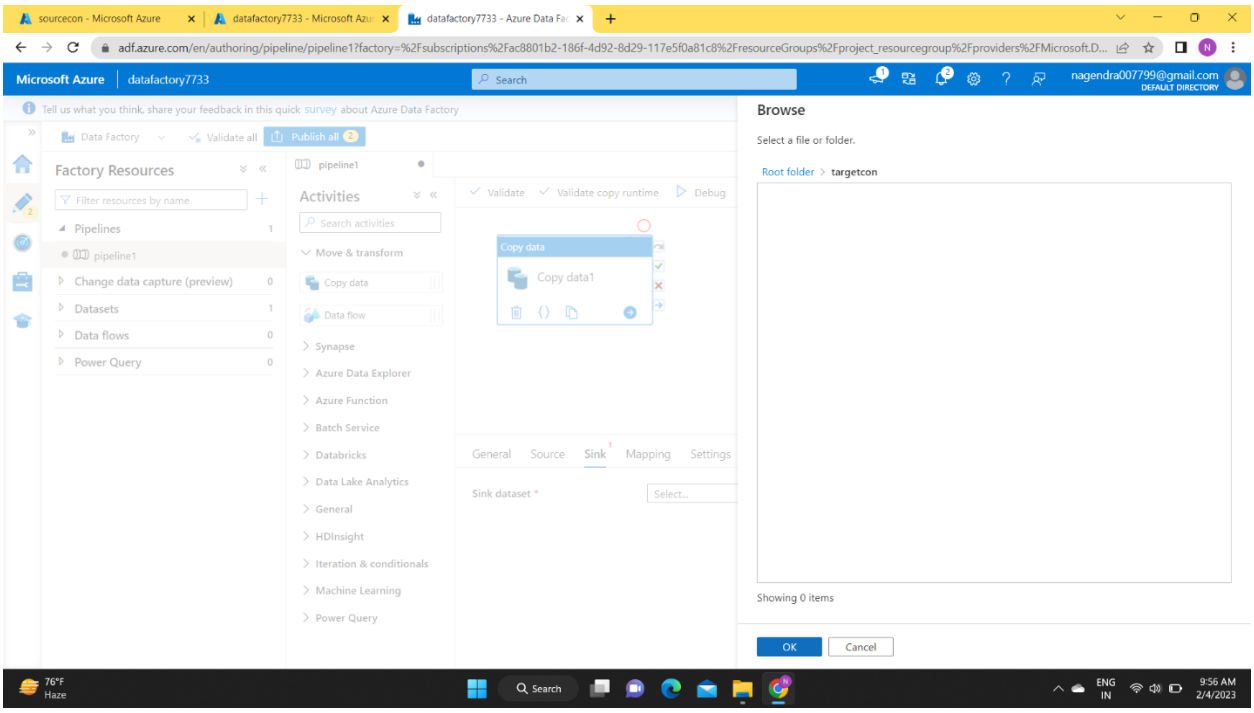


- We have to give target “file path” where we want to store our target file.

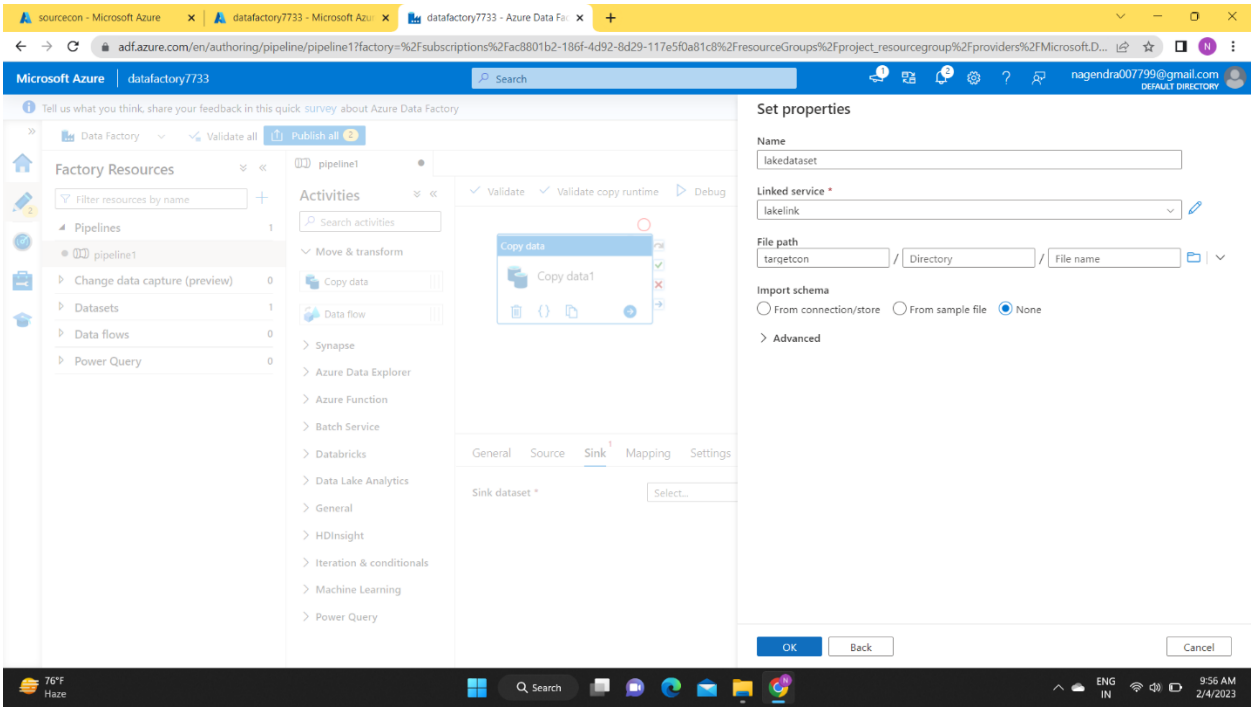




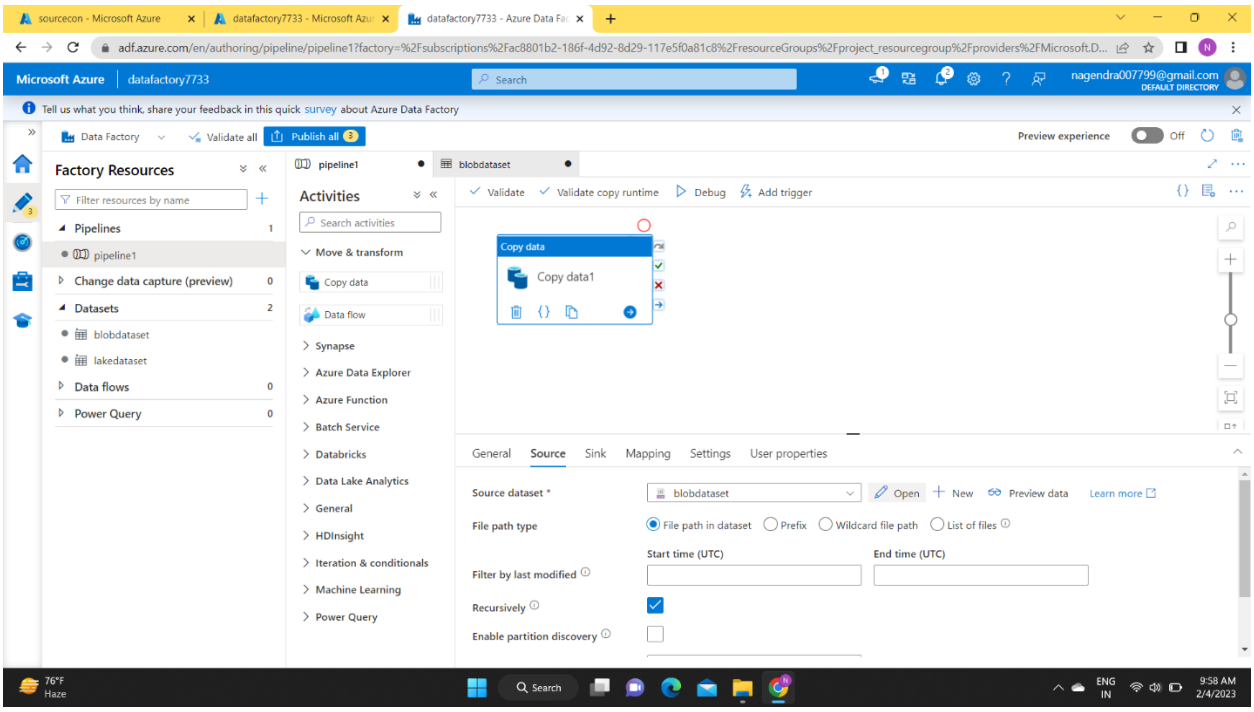
- Here no files in our target it's empty.



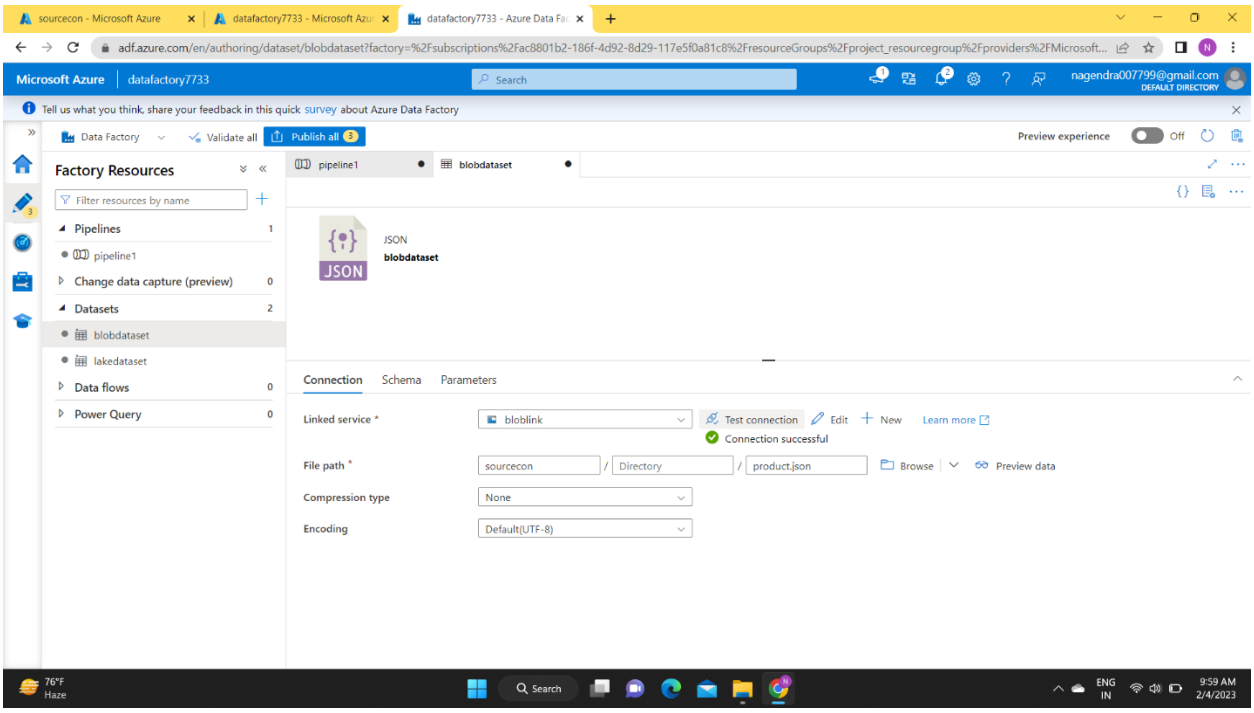
- We are not importing schema so we put none.



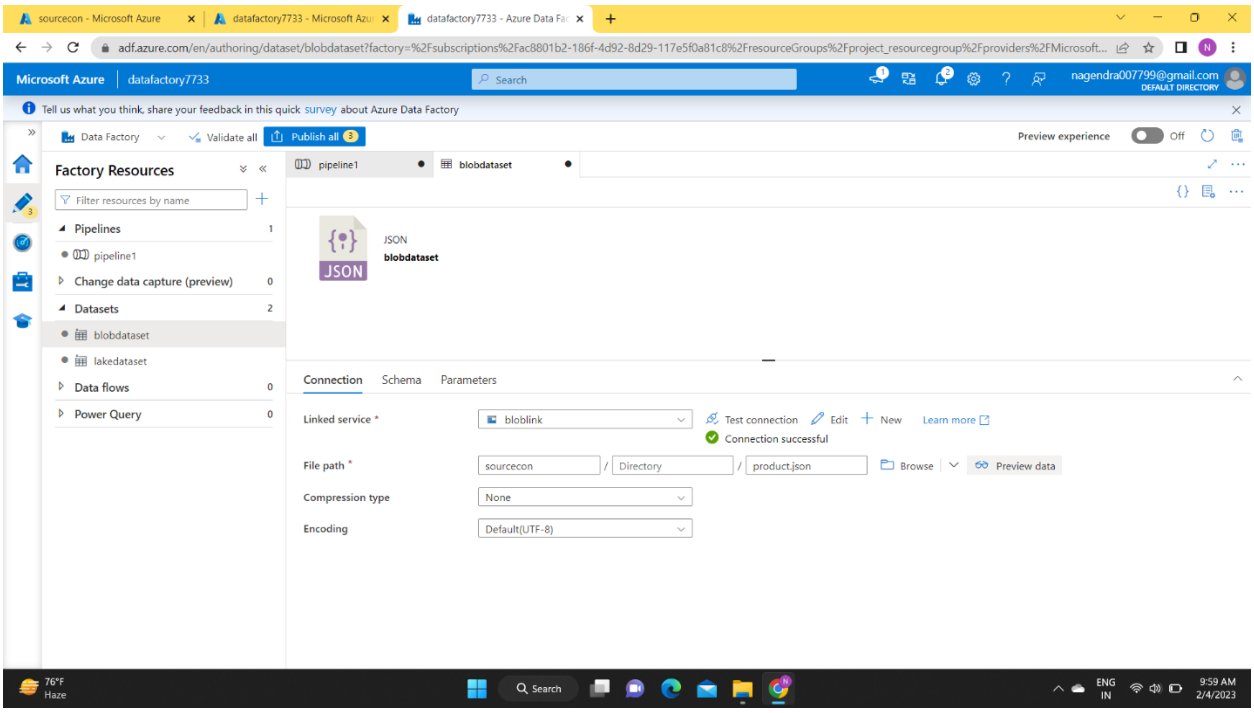
- In source option 2nd row “file path type” in that row select “file path in dataset”.
- We are transferring single file so we select “file path in dataset”
- If you want to transfer multiple files select “wild card file path”.
- Go to source there is an “open” option with pencil mark.



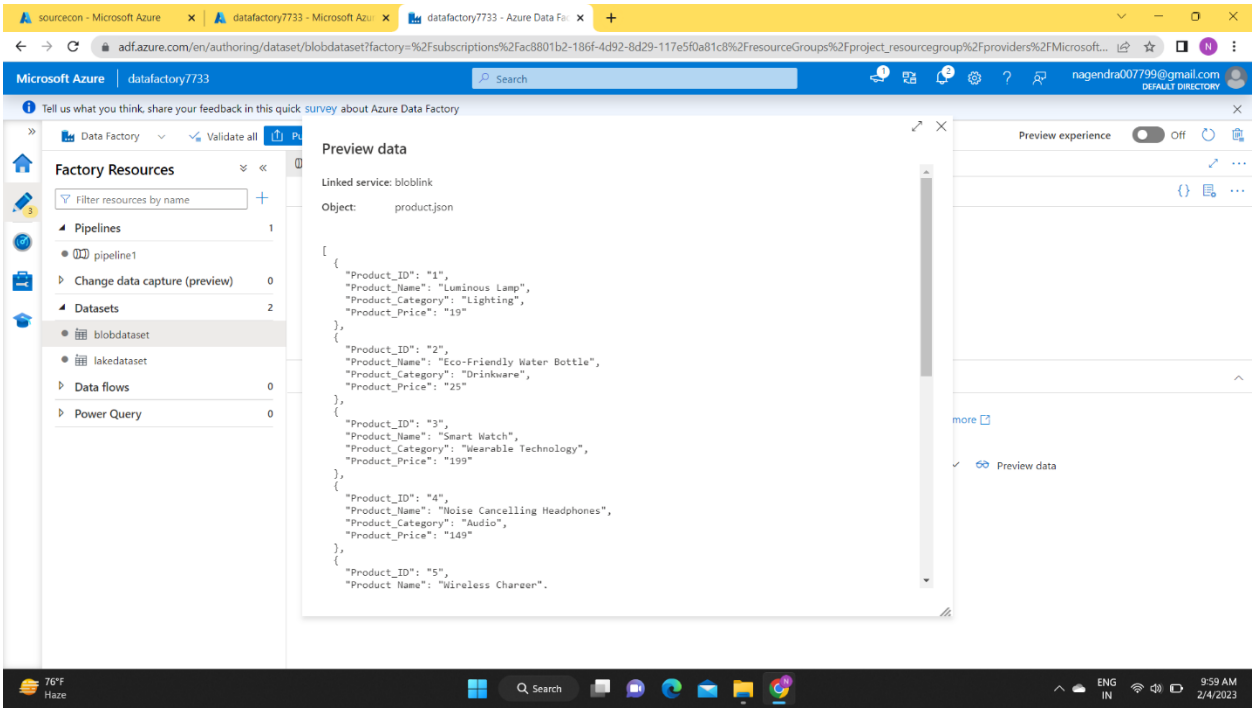
- Must do the test connection.



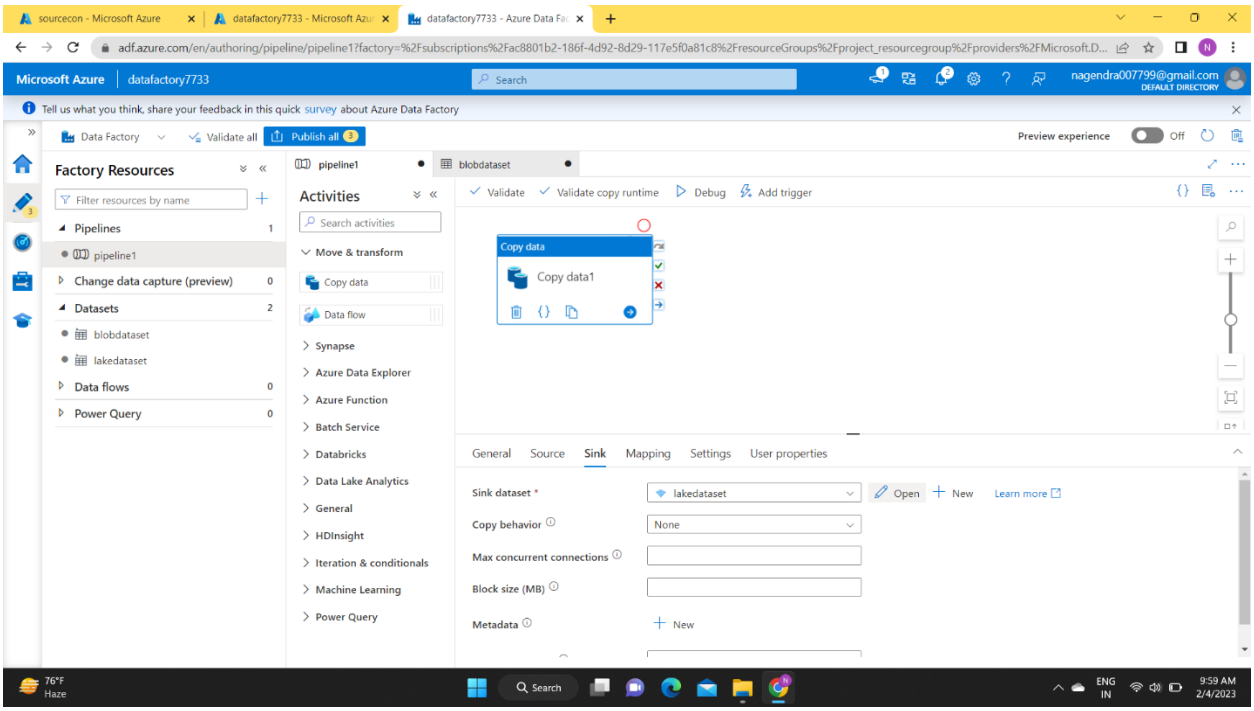
- Then click on preview data.



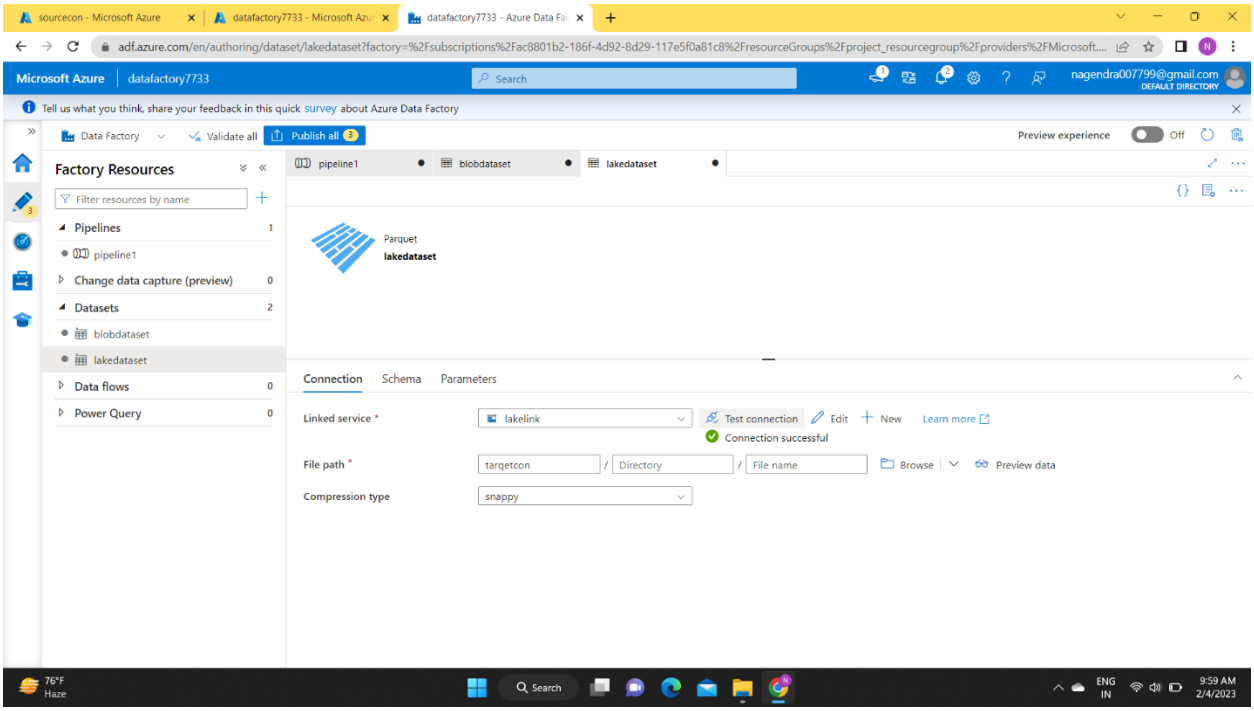
- This is our source json file format product data preview.



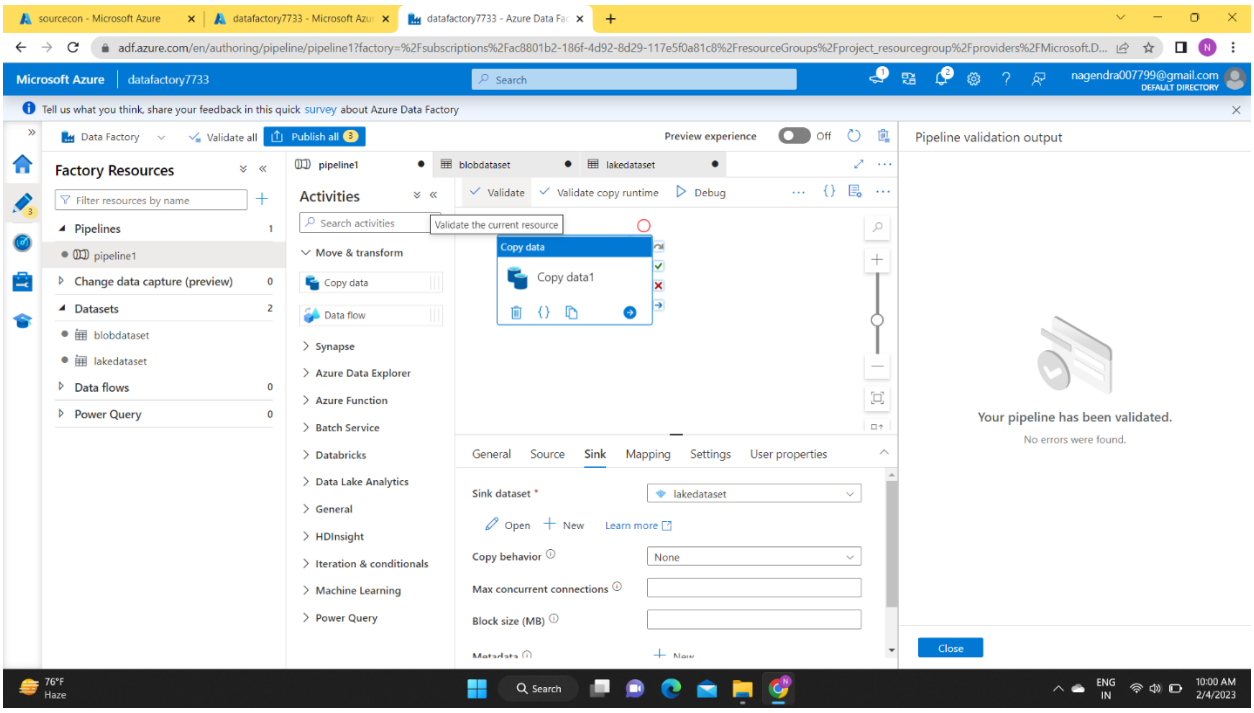
- Now go to sink there is “open” option.



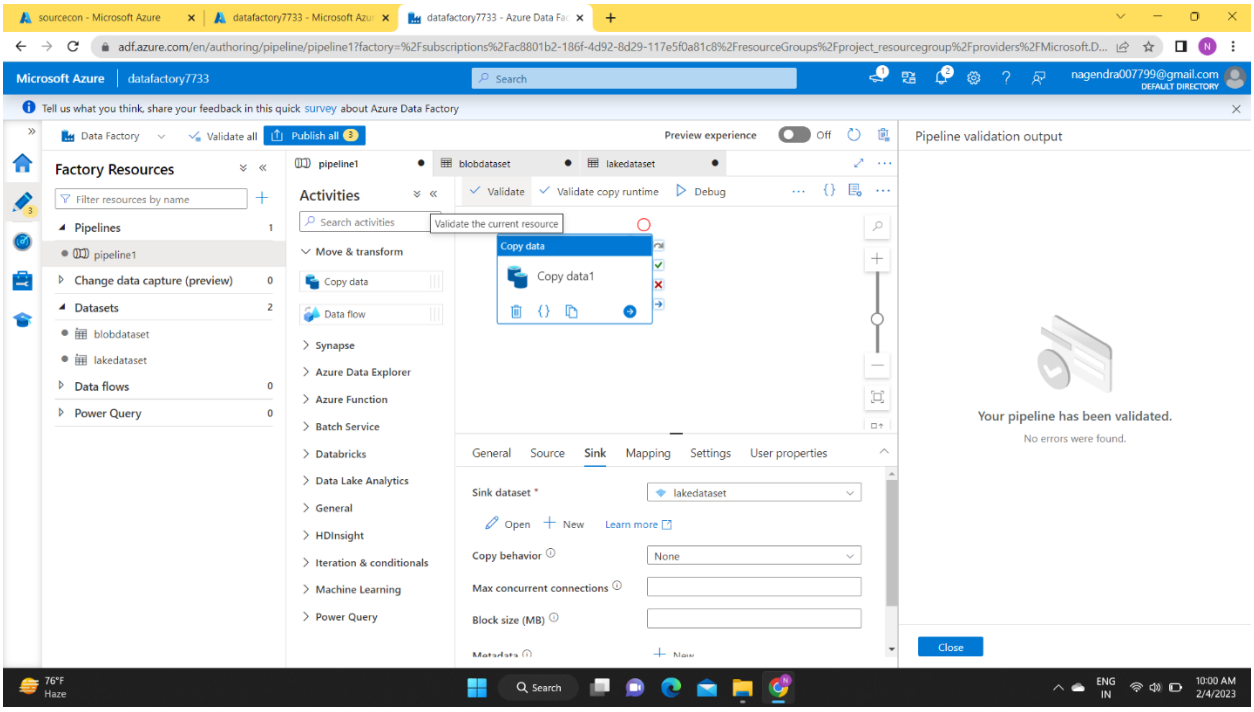
- check the test connection



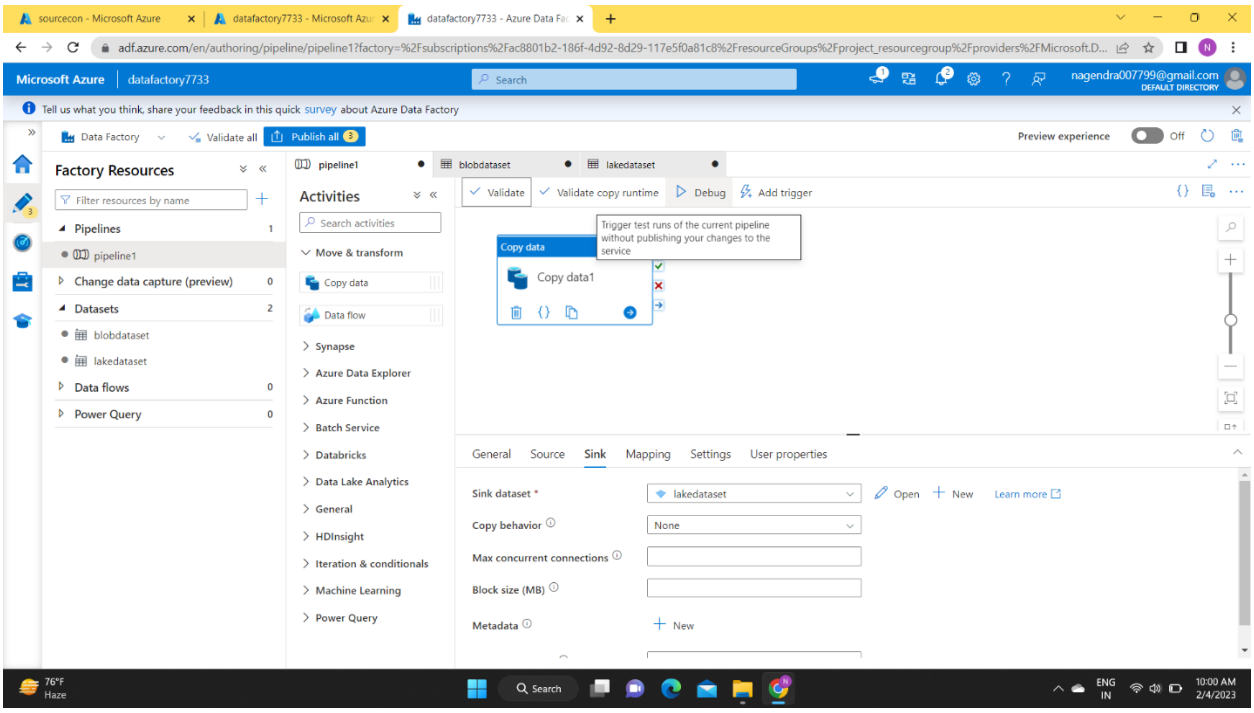
- now validate your pipeline it will show errors on right sidebar



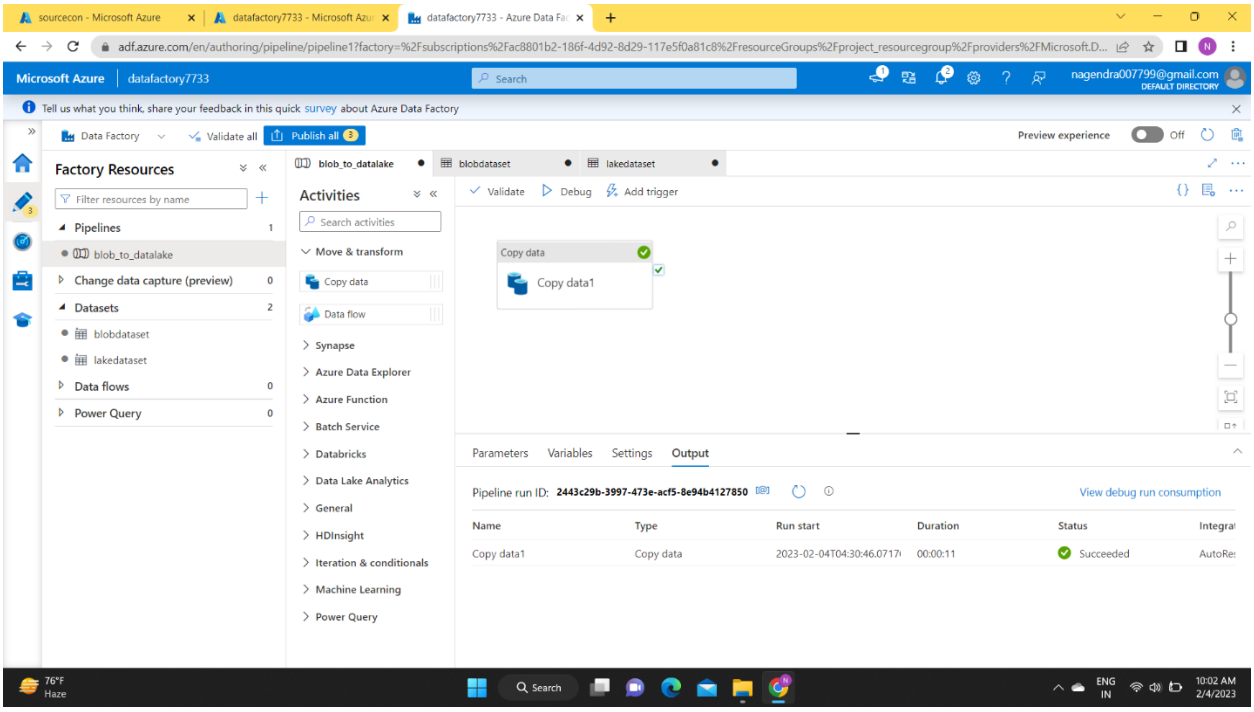
- our validation complete we don't have any errors.



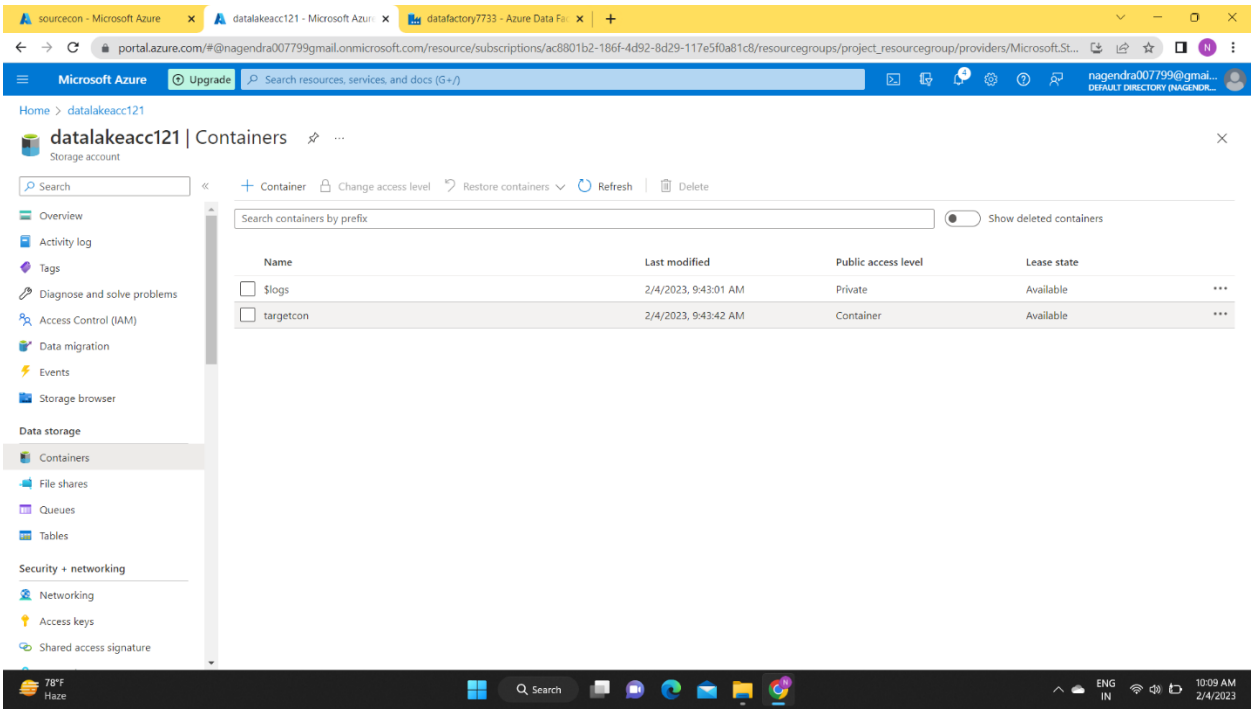
- now do the debug



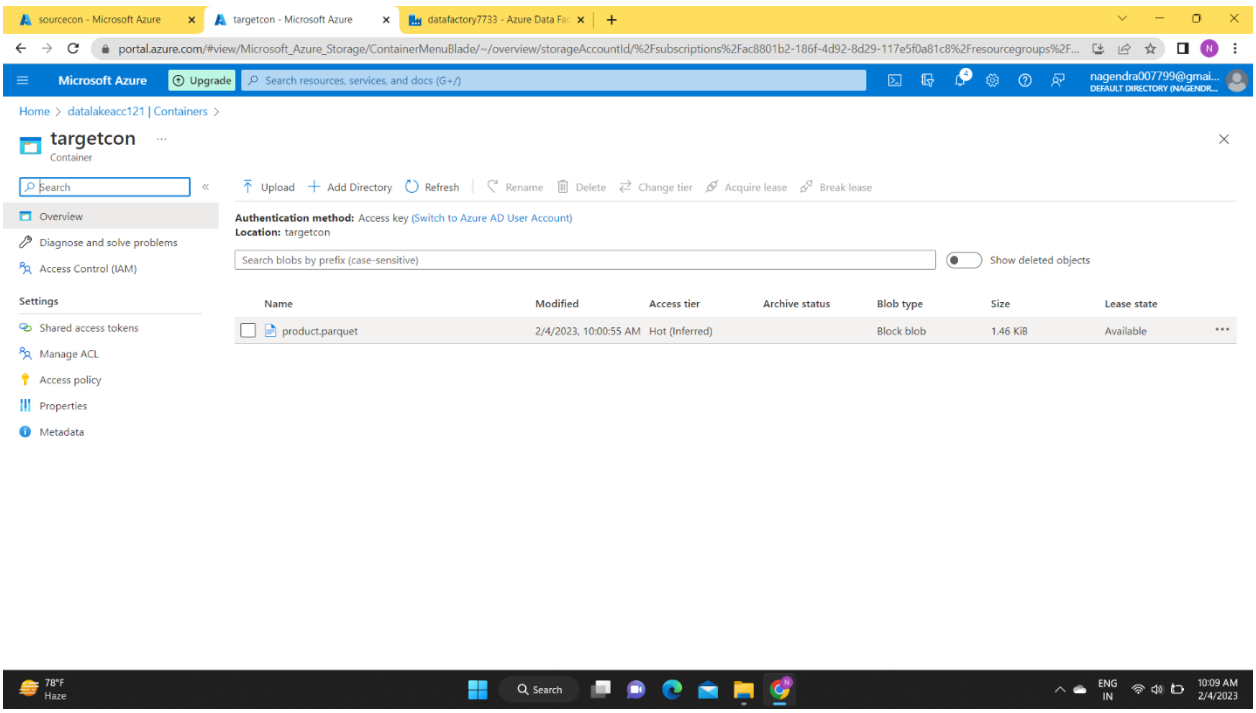
- our pipeline is succeeding, and you can see that in message box in pipeline.



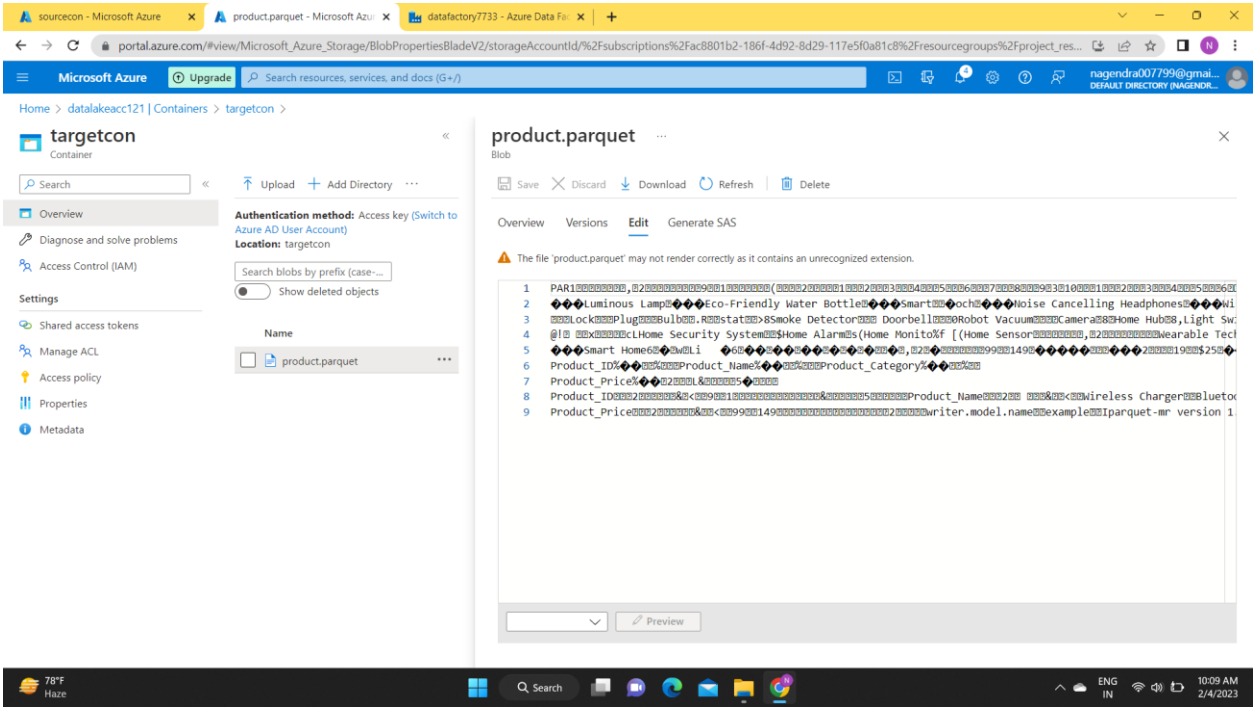
- to check copy data we are going to data lake.
- In data lake, targetcon our parquet file should be present.



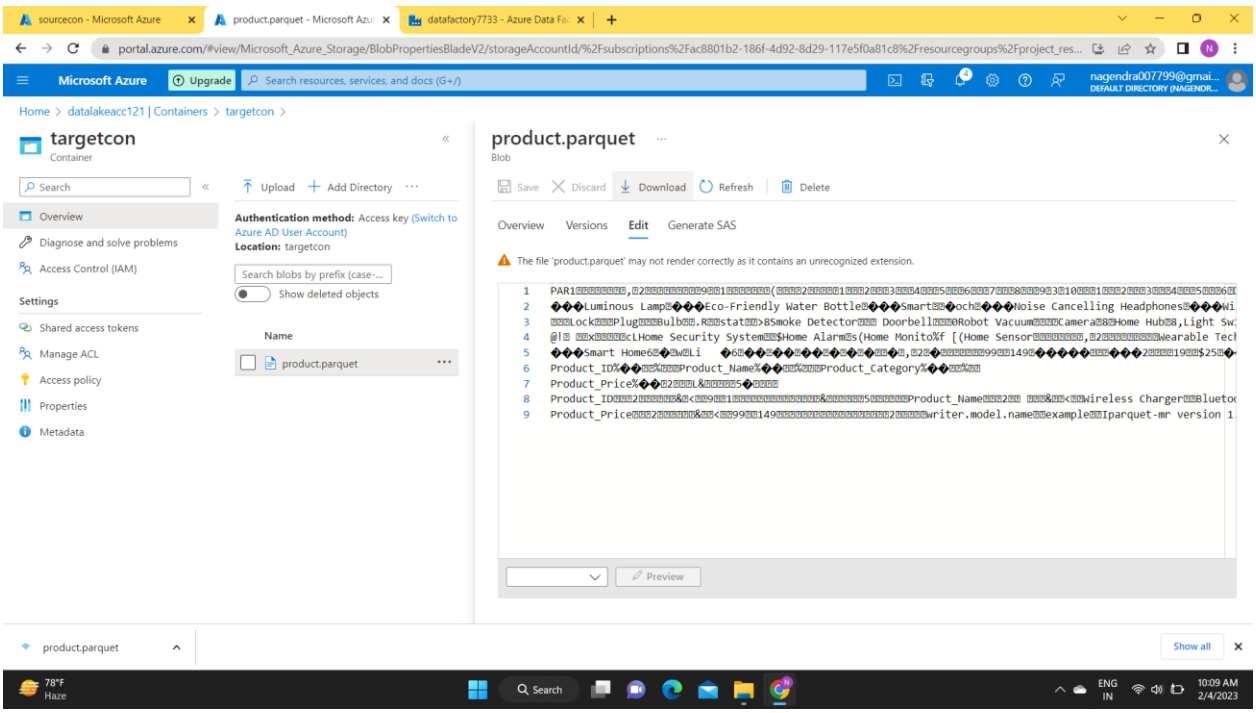
- Here is our parquet file in datalake



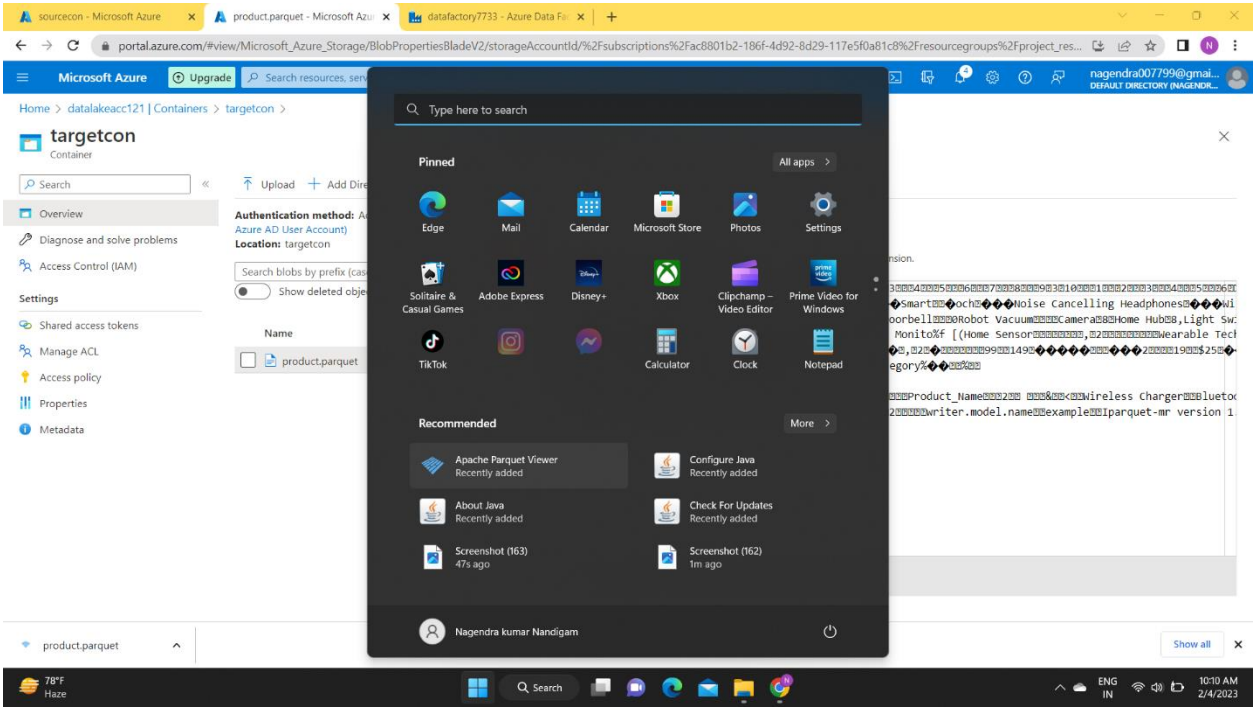
- Click on “edit” to see data present in file



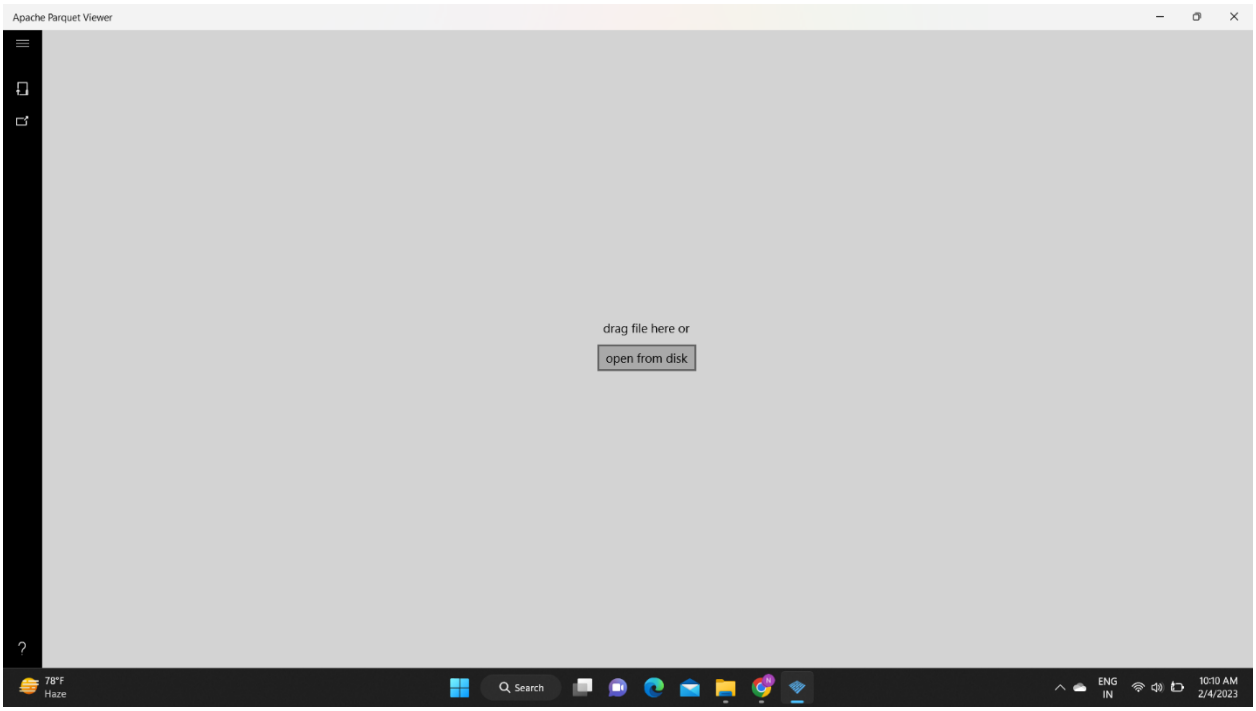
- We cannot understand the file so we have to download file.

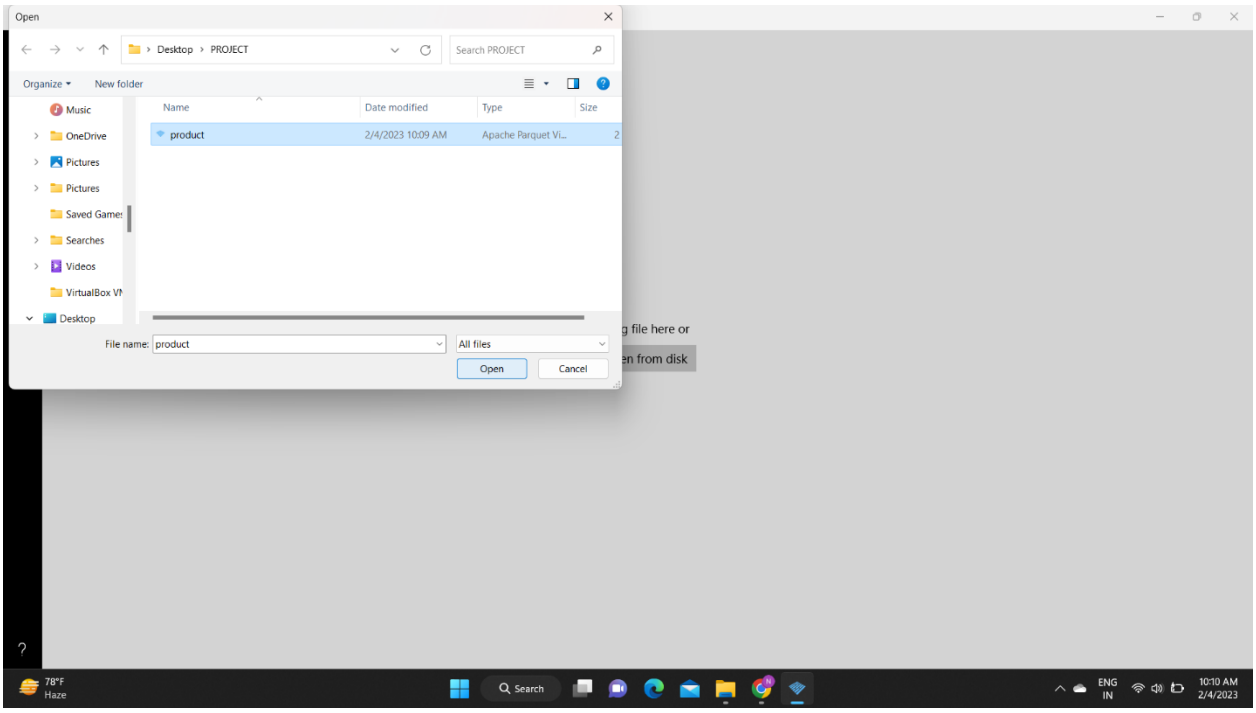


- You have to download the advanced parquet viewer app to see data in parquet.

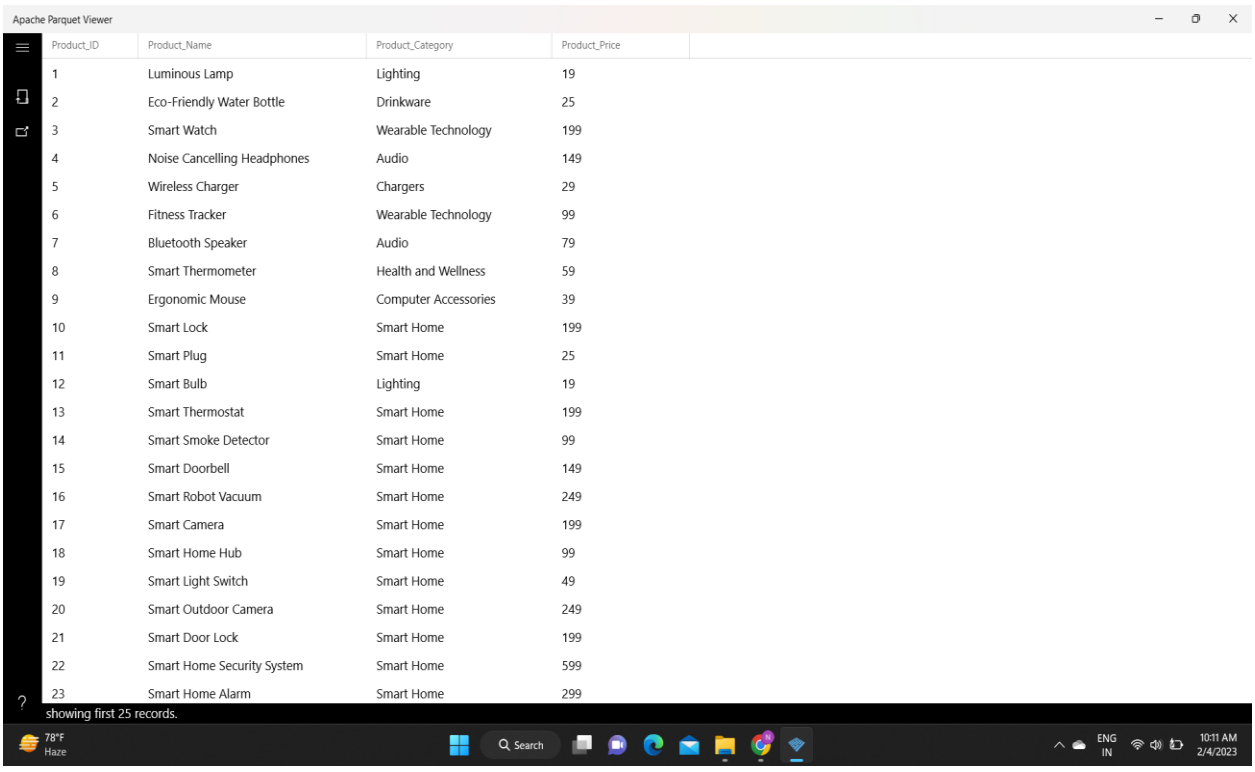


- Now open the advanced parquet viewer app drag the downloaded file from your pc.

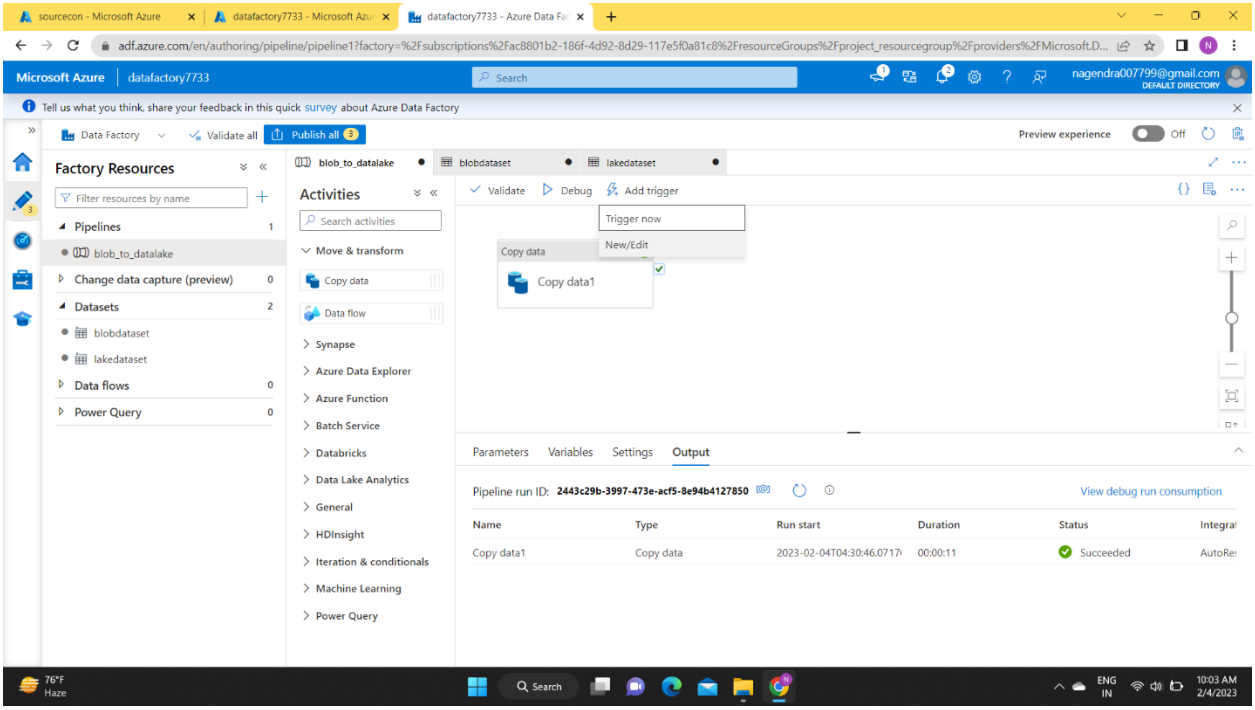


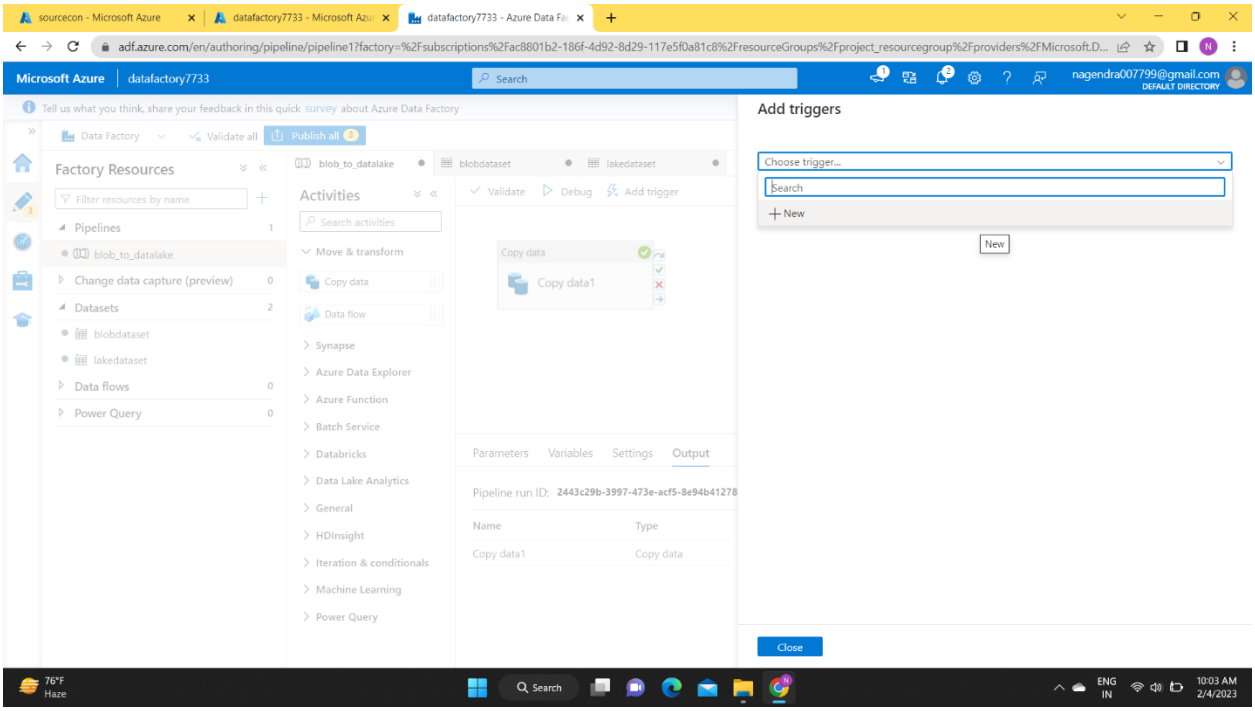


- Now the product data is displayed.

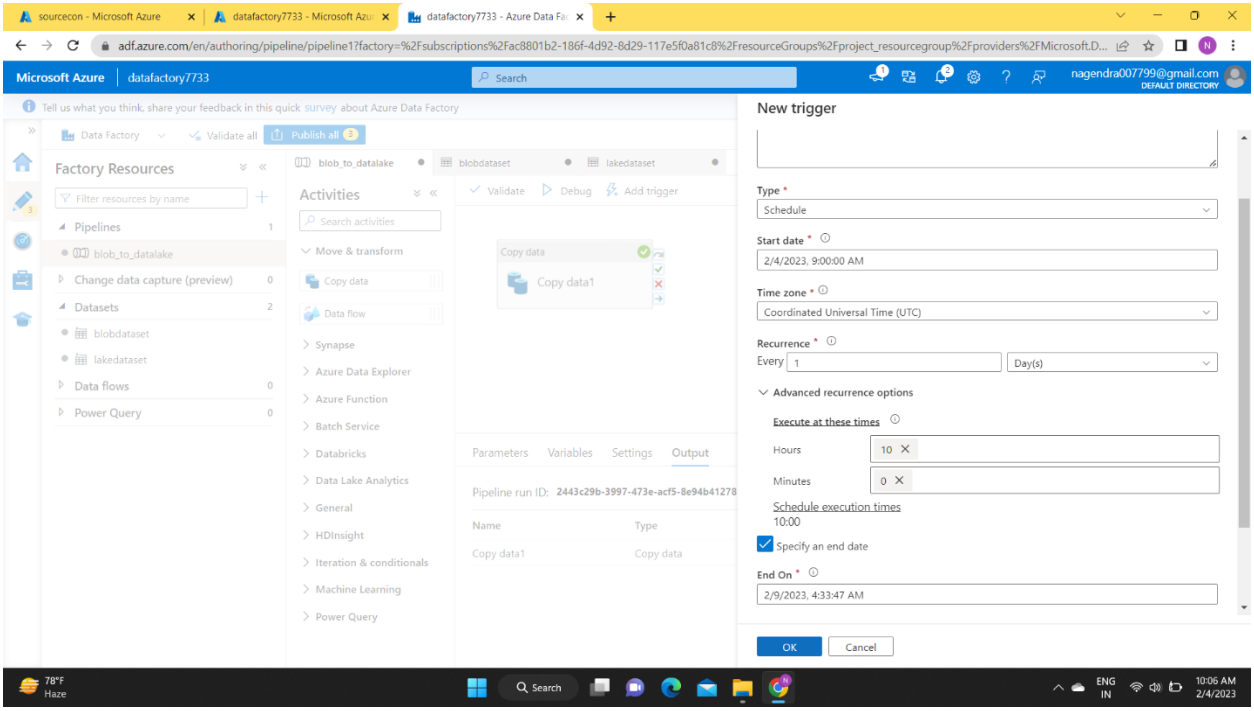


- Now we are set up the trigger pipeline by clicking “add trigger” and “new/edit”

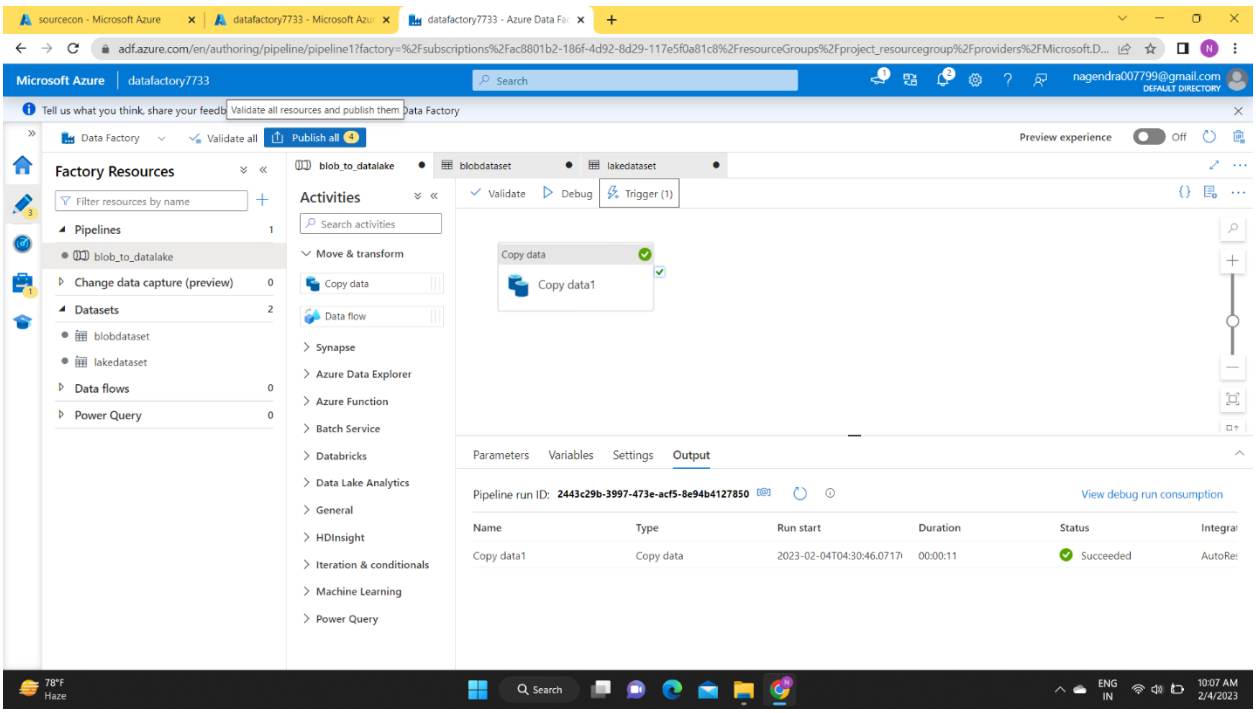




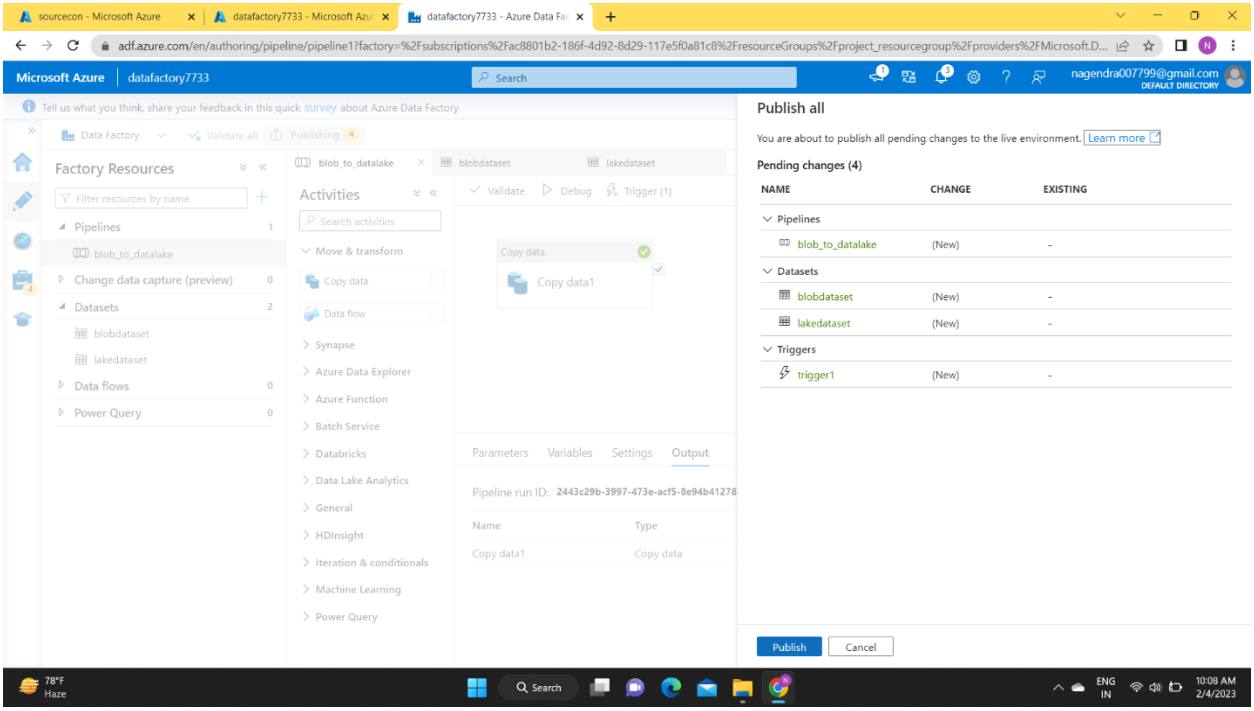
- Here I am using “schedule trigger”



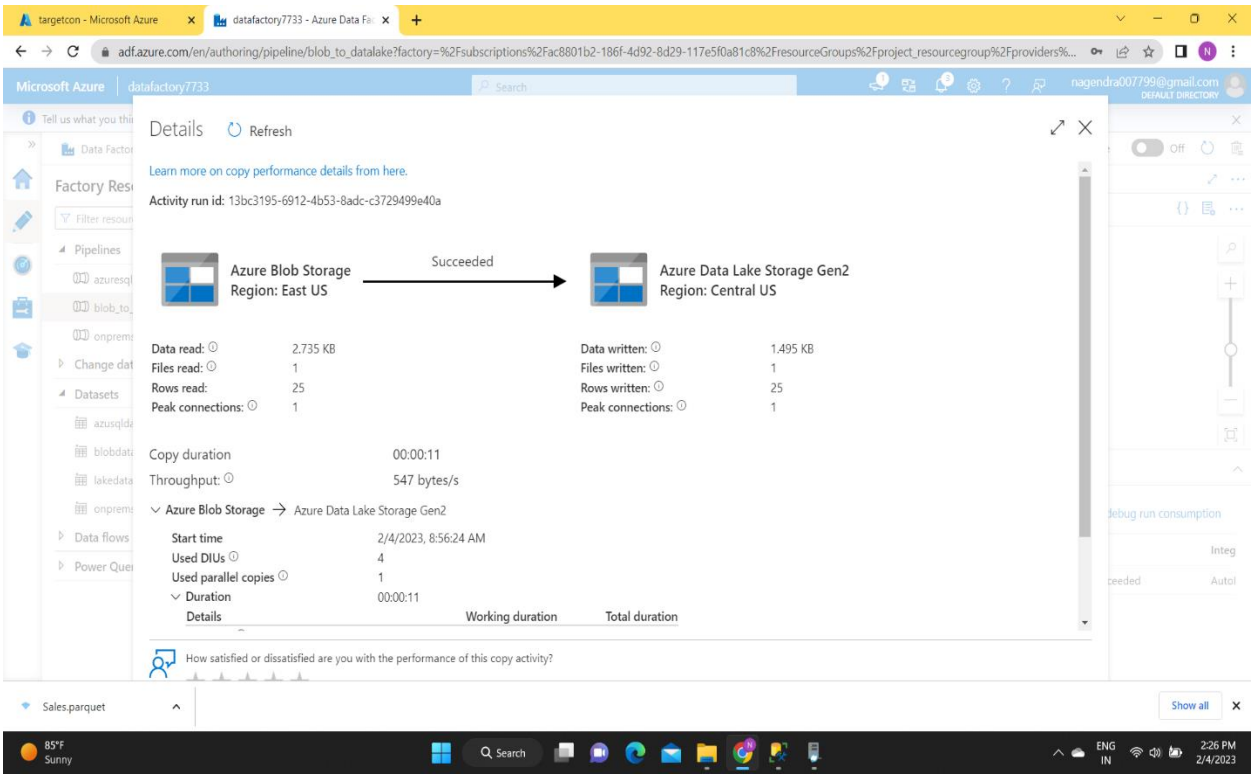
- Now I am attaching trigger successfully.



- Now I am publishing the pipeline with trigger.

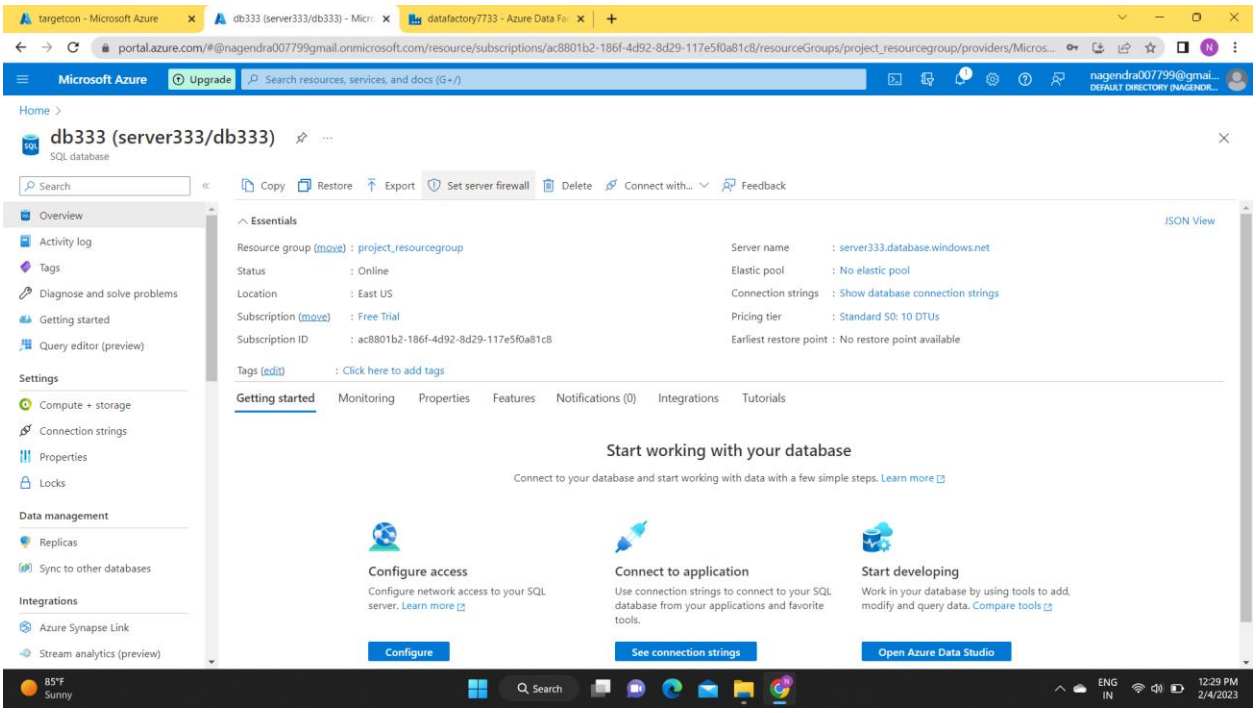


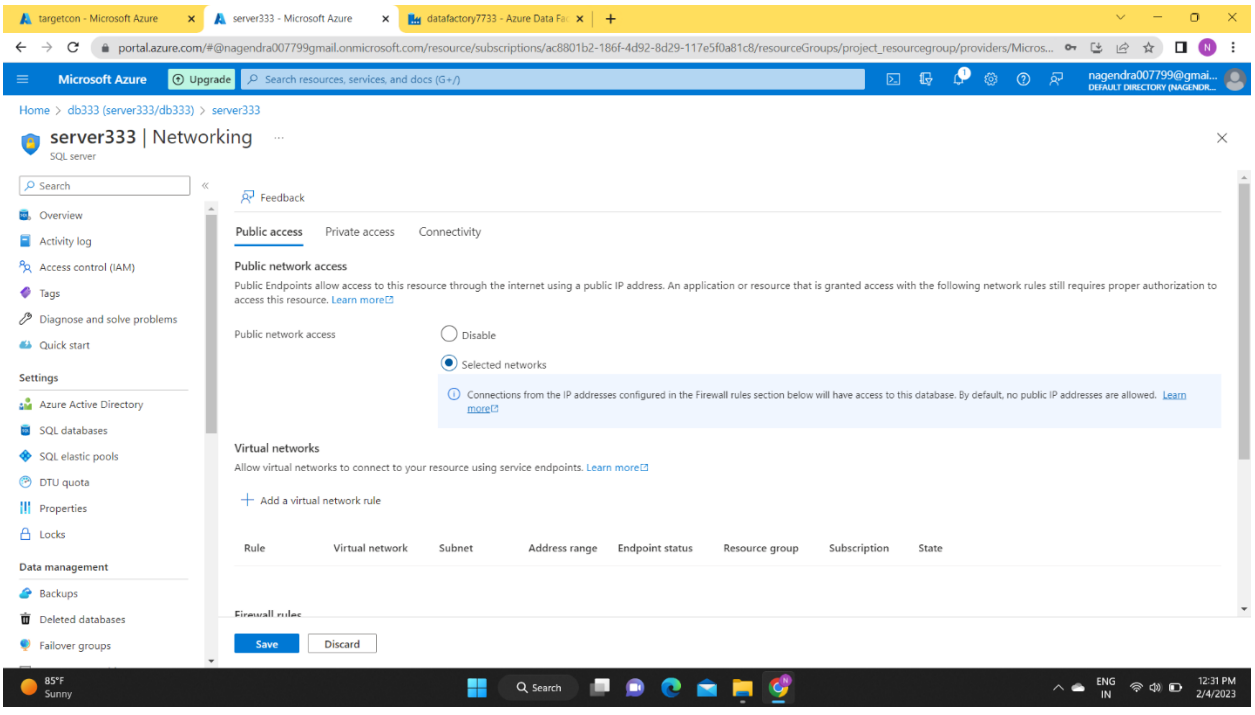
- This is the pipeline we are done.

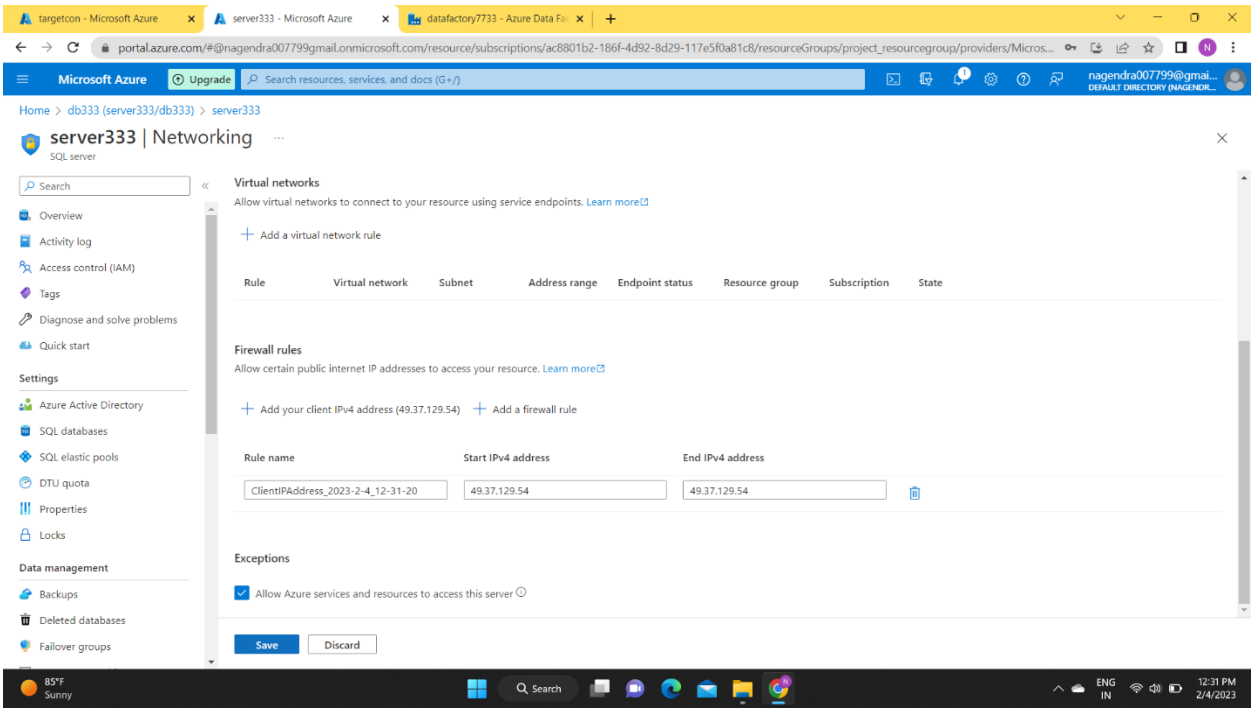


Copy data from azure sql db to azure data lake

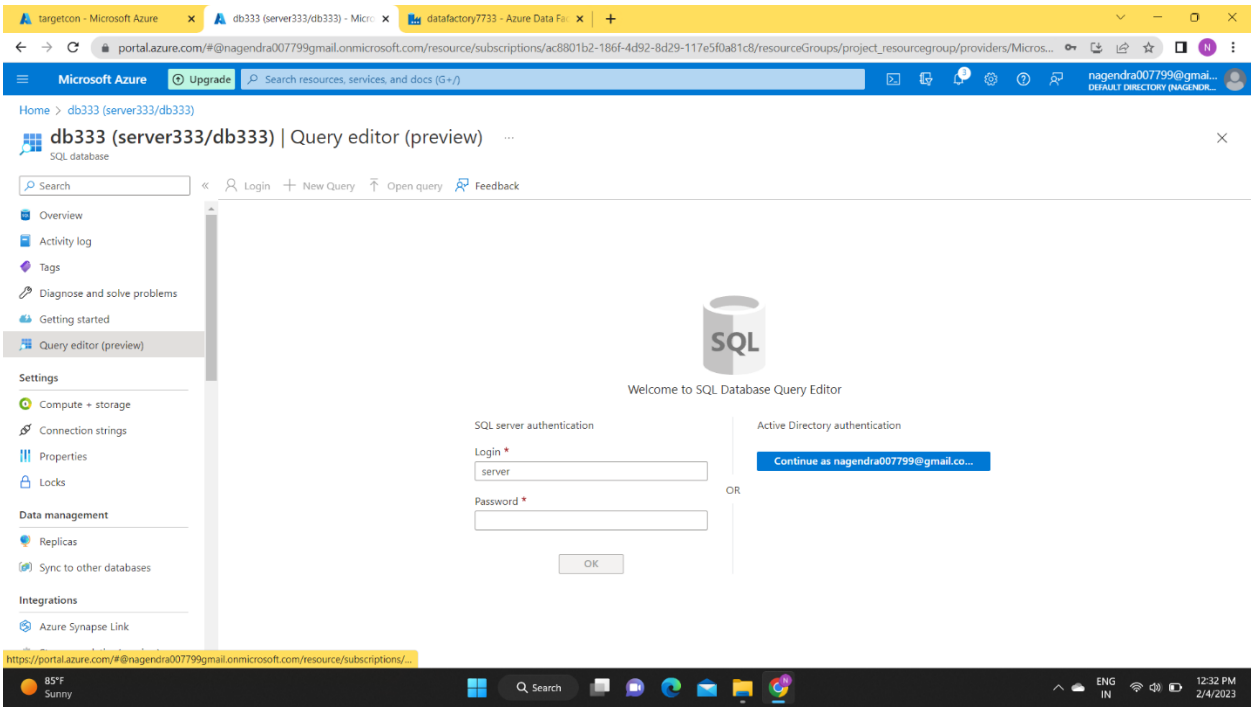
- This is our azure sql database “set server firewall” to give access to our azure resources to this sql db



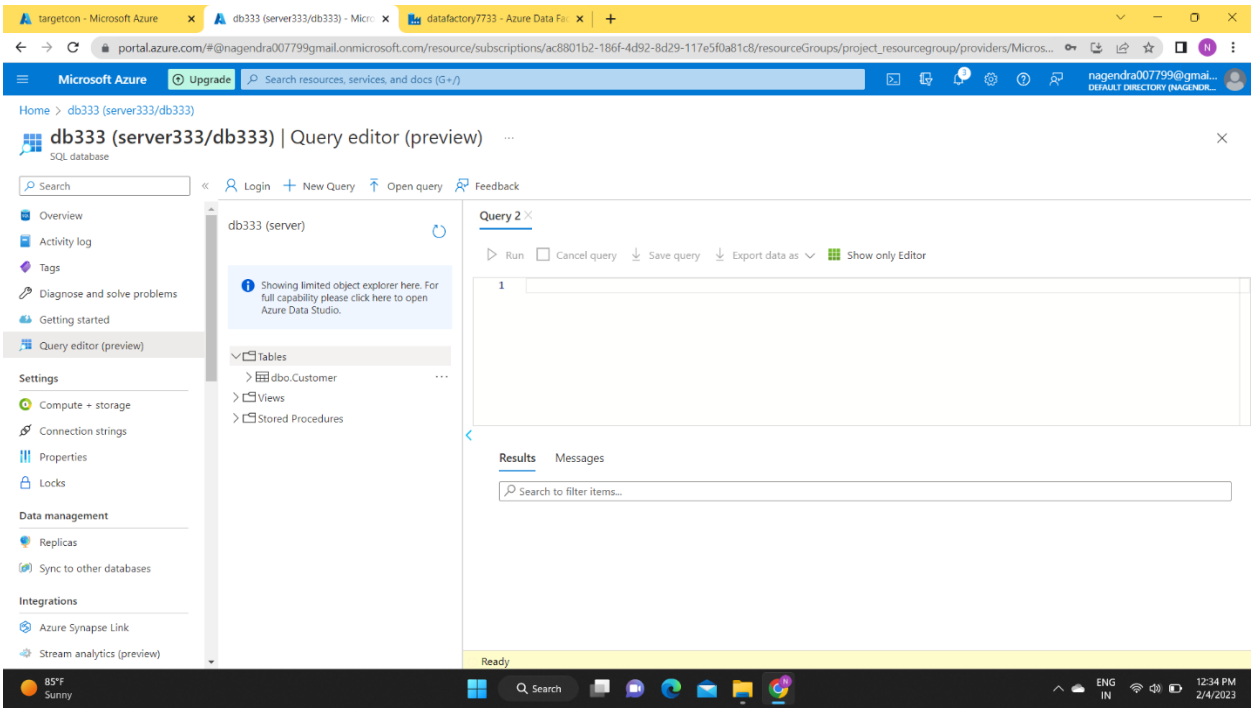




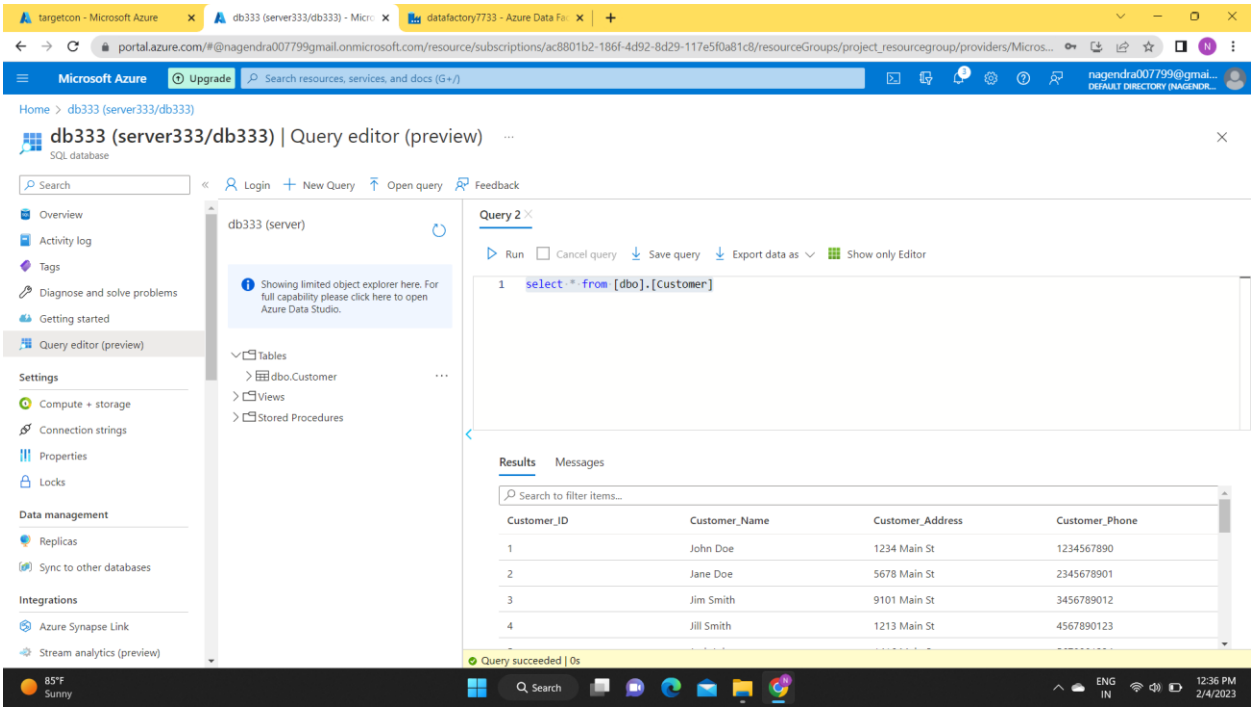
- You can login azure sql db by using “query editor” option in left side of azure sql portal.



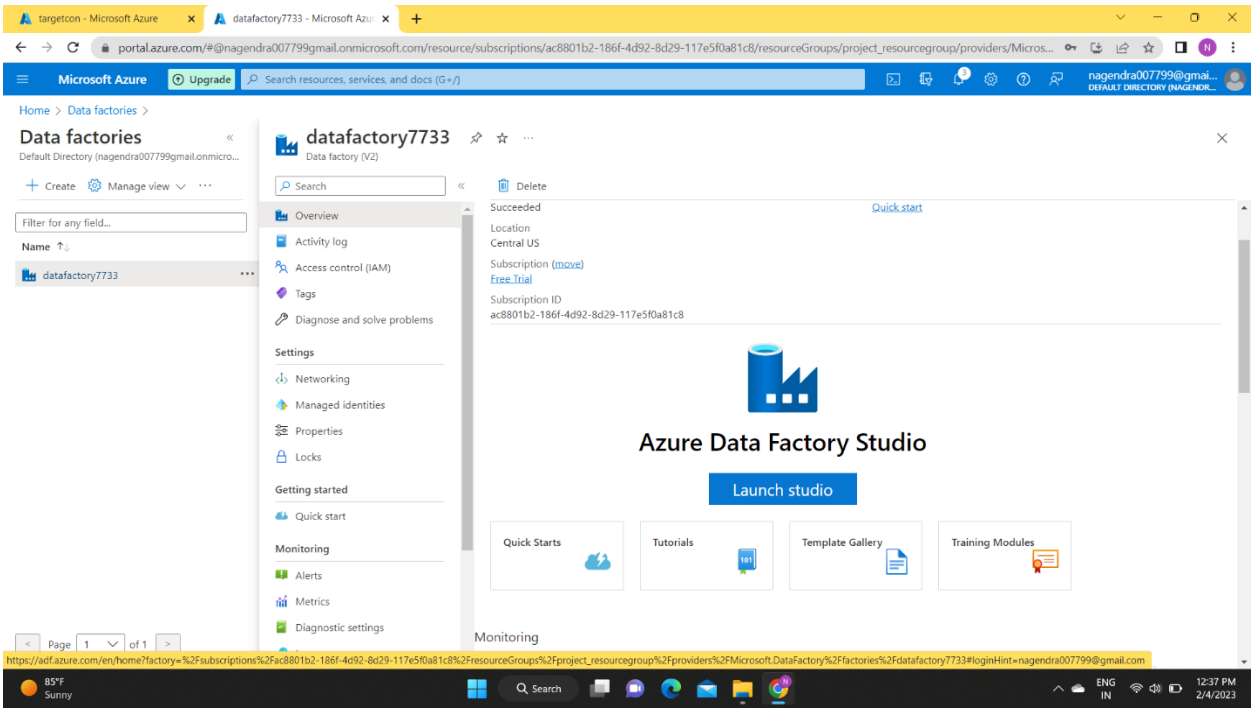
- Here you can see our customer table “dbo.customer”.



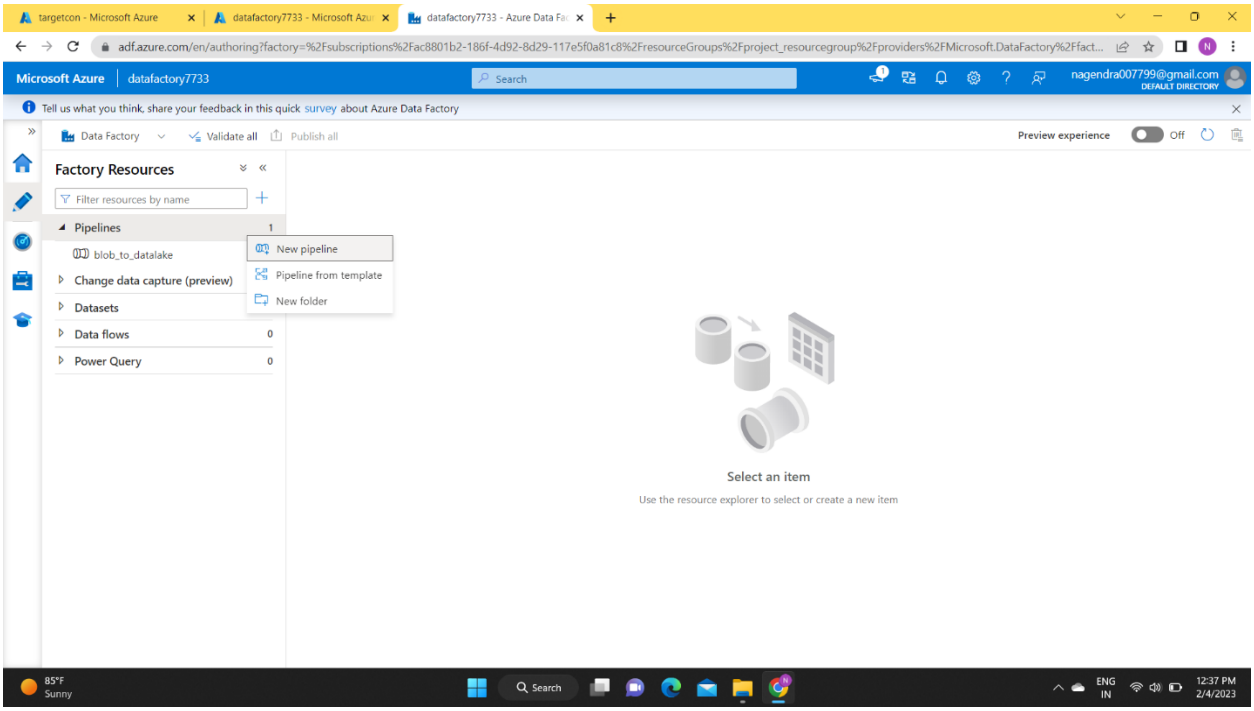
- You can see the data in customer table.

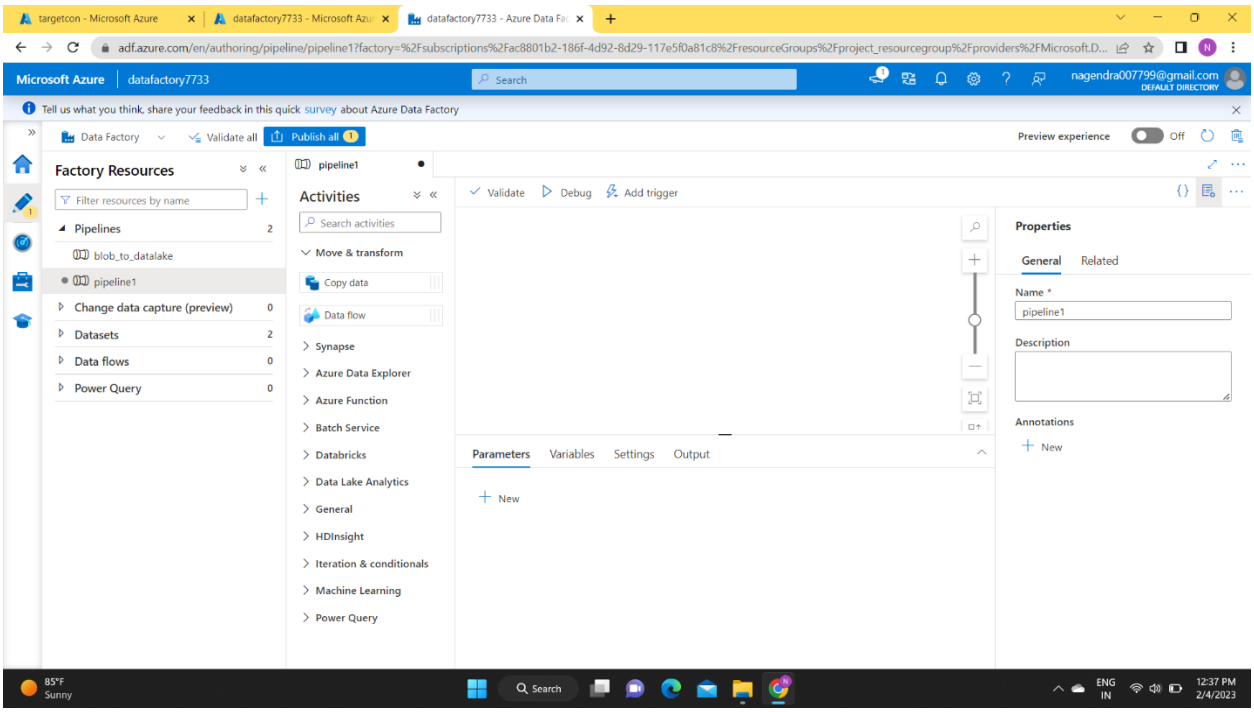


- Now we can launch azure data factory.

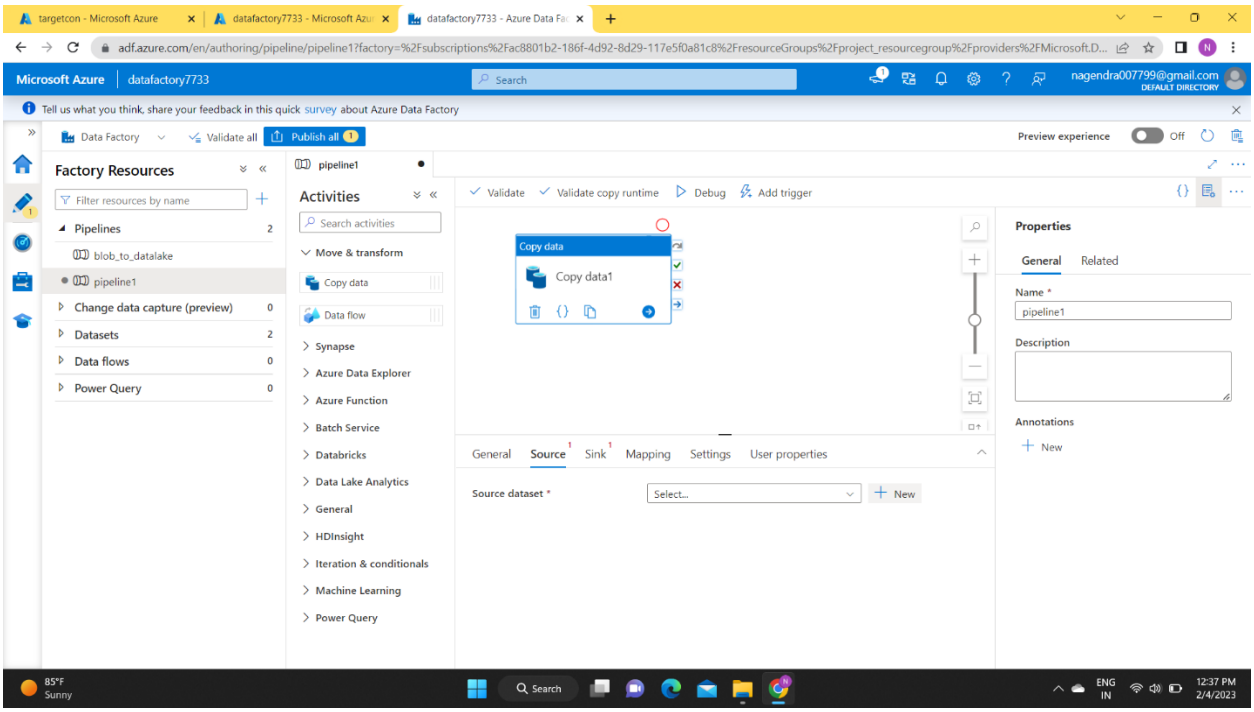


- Create new pipeline

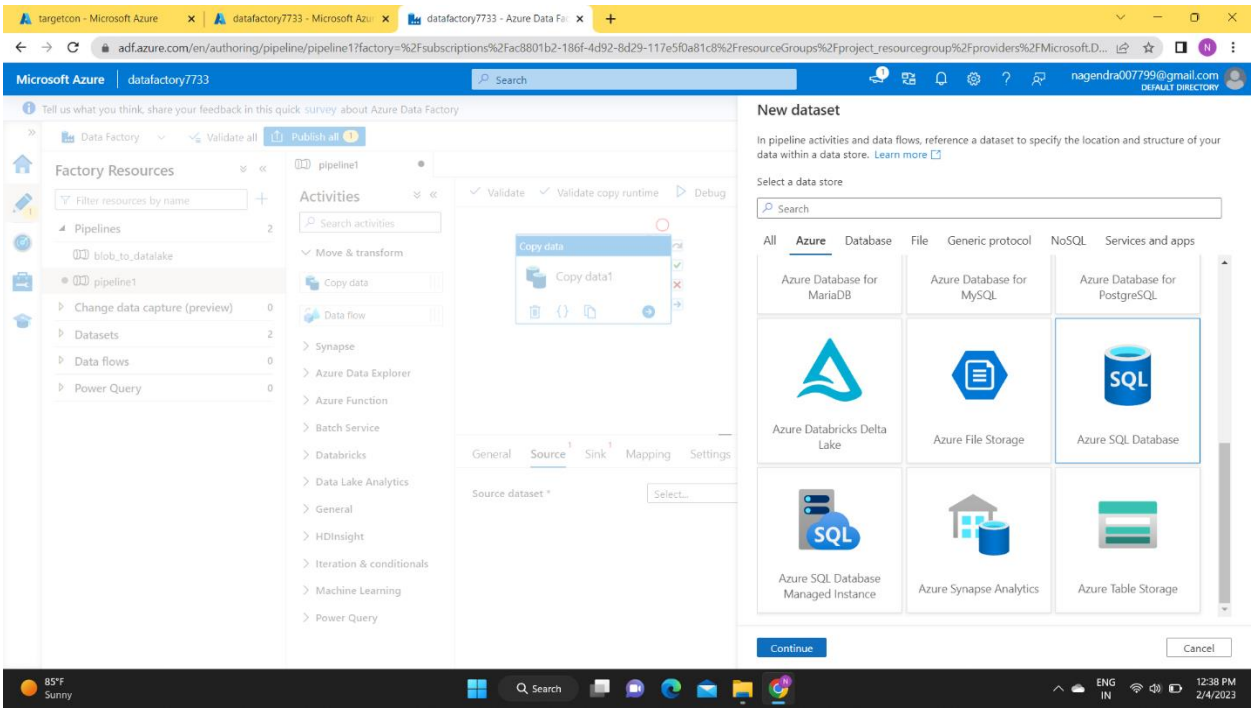




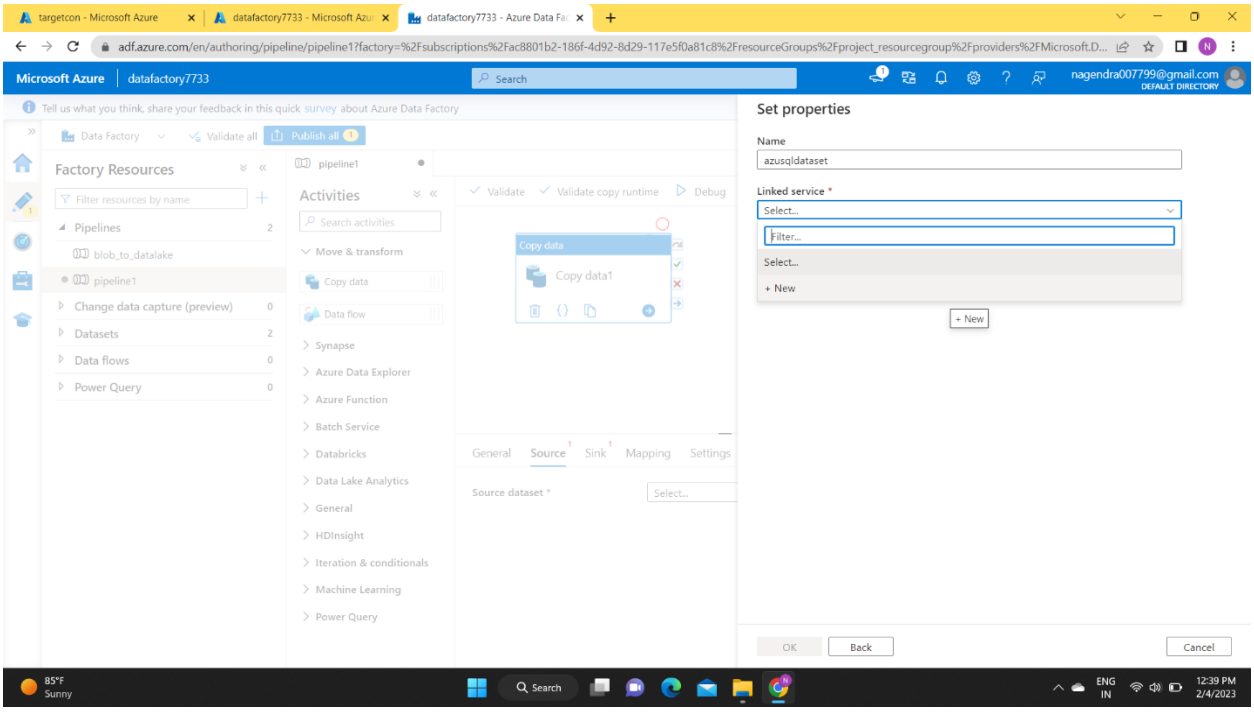
- Select copy data activity in adf.
- In source create new dataset for source azure sql db.



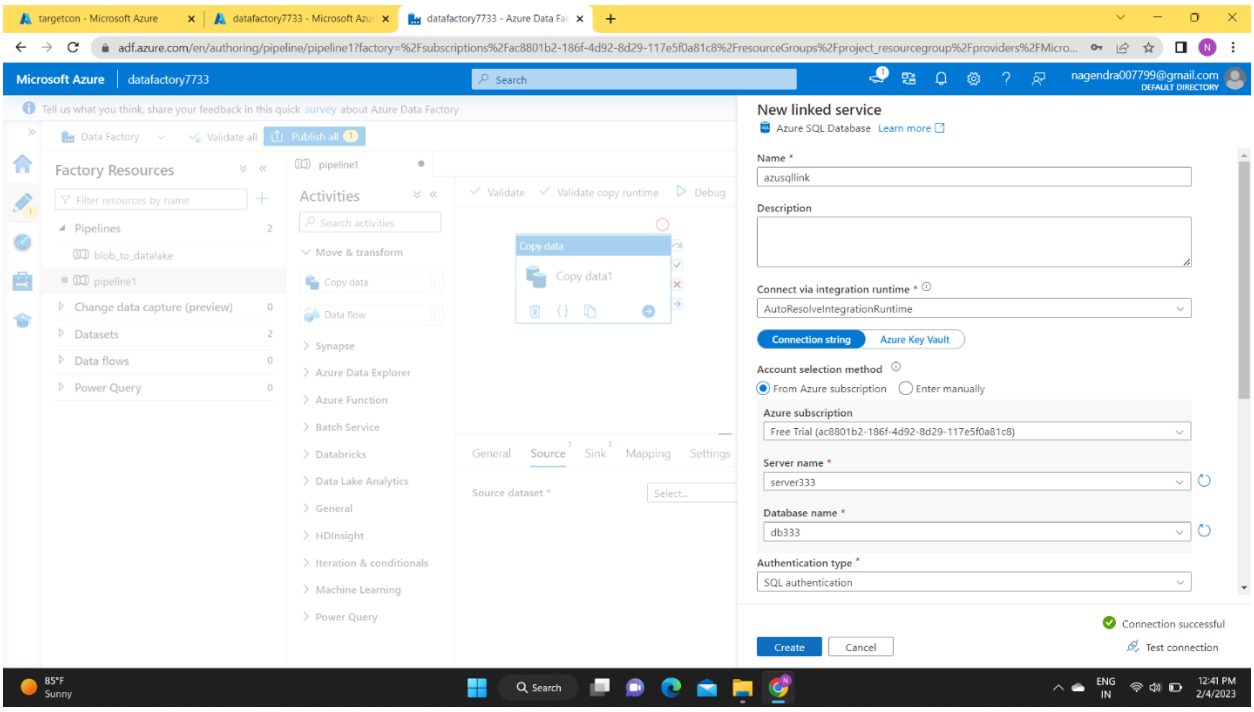
- In source you can choose azure sql db.



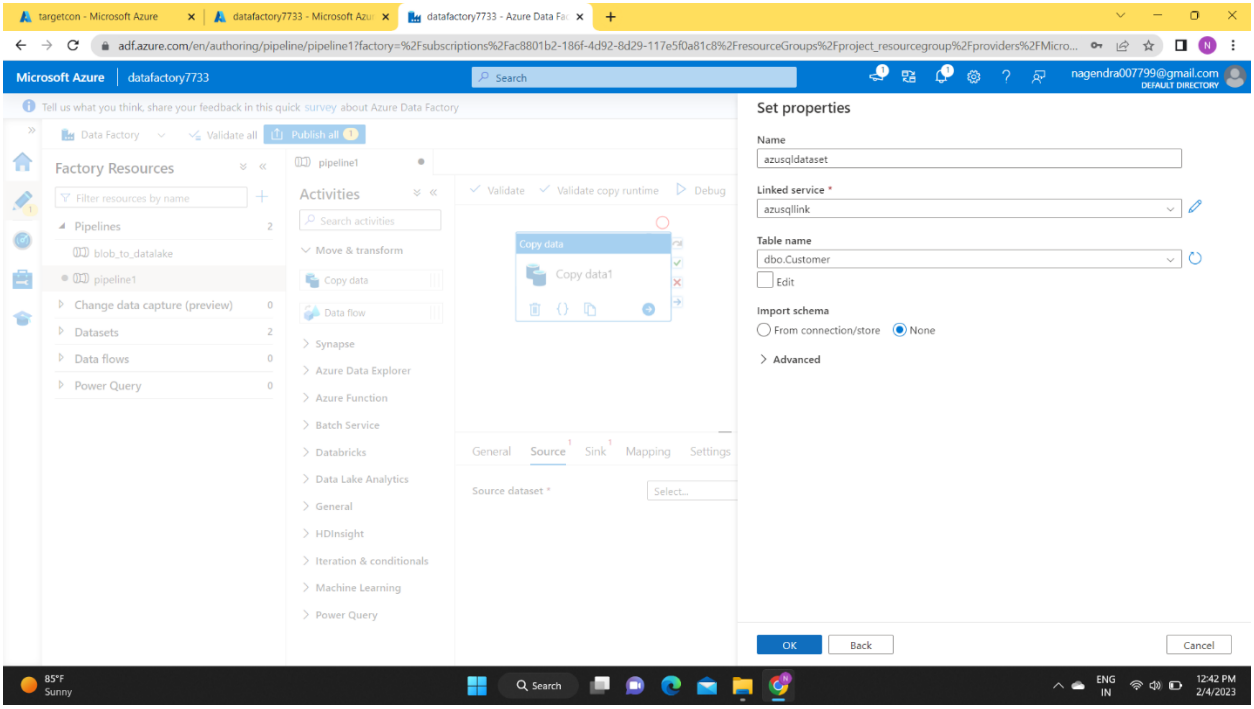
- Our dataset name is “azusqldataset”.



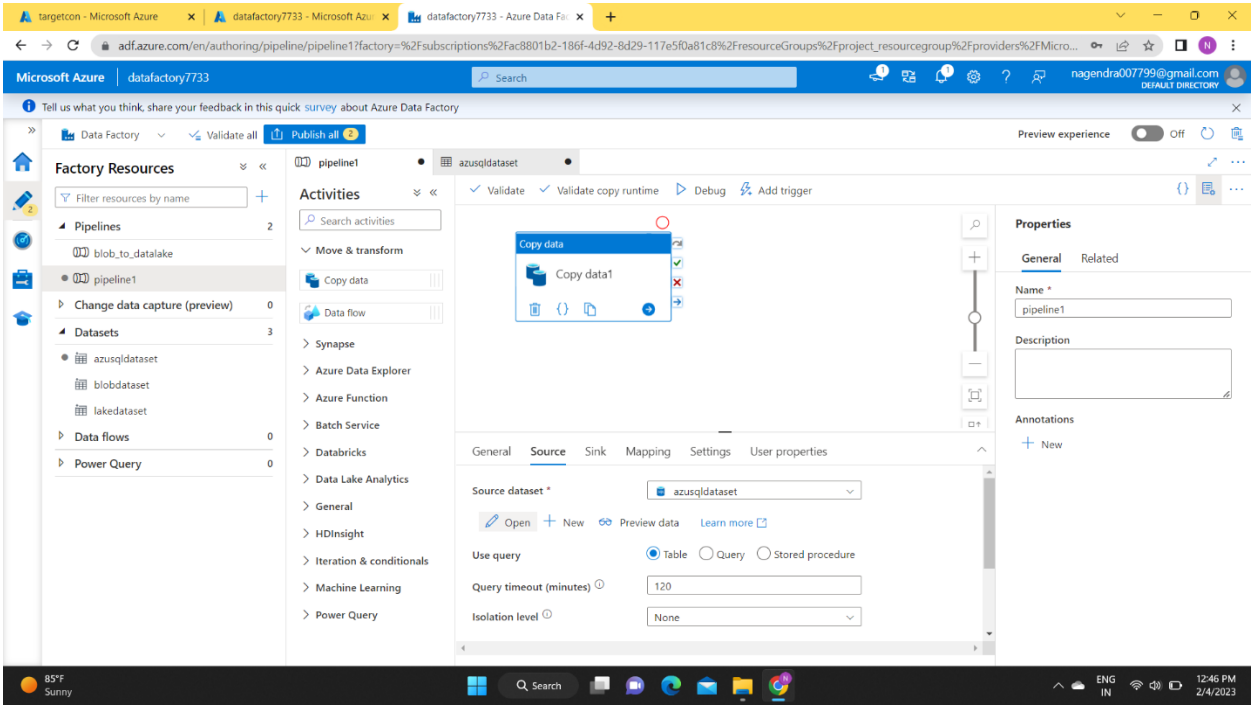
- Our azure sql dataset link service name is “azusqllink”
- Here we have to create sql server.
- The name of sql server is “server333”
- Test the connection.



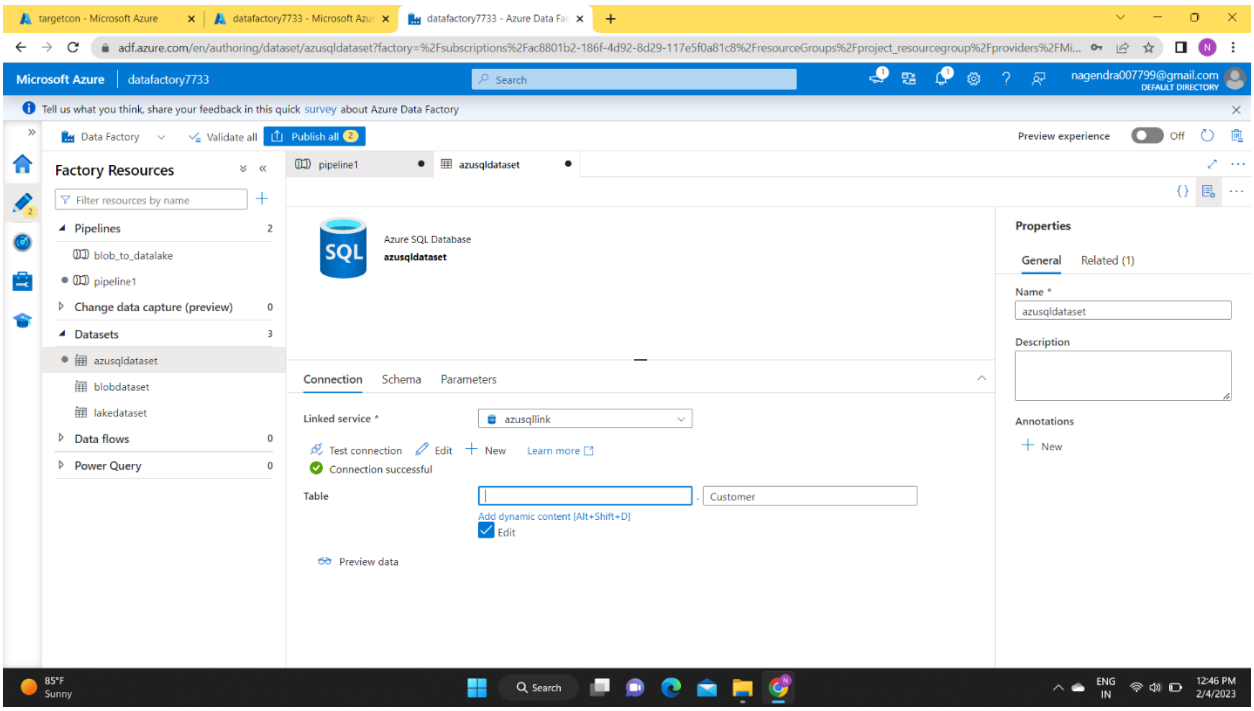
- You select your table and put import schema should be none because we are importing data from .sql to parquet file format.



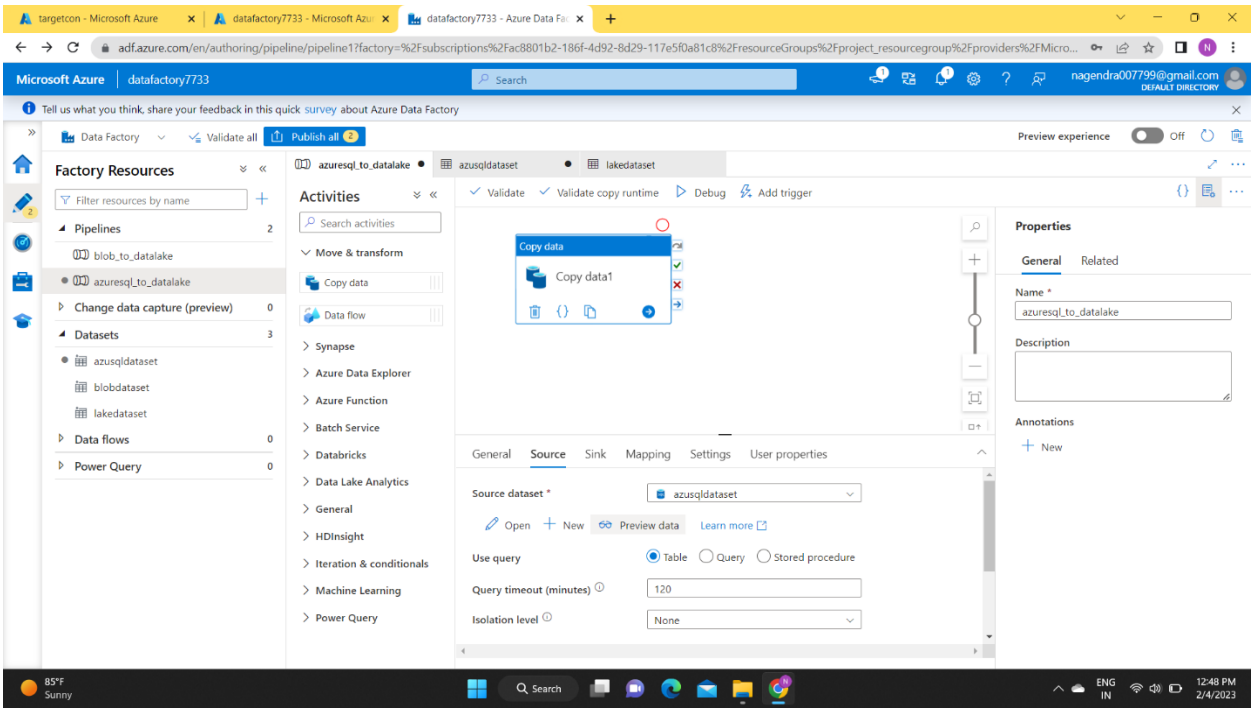
- Insource we are selecting “open” to edit our table name.



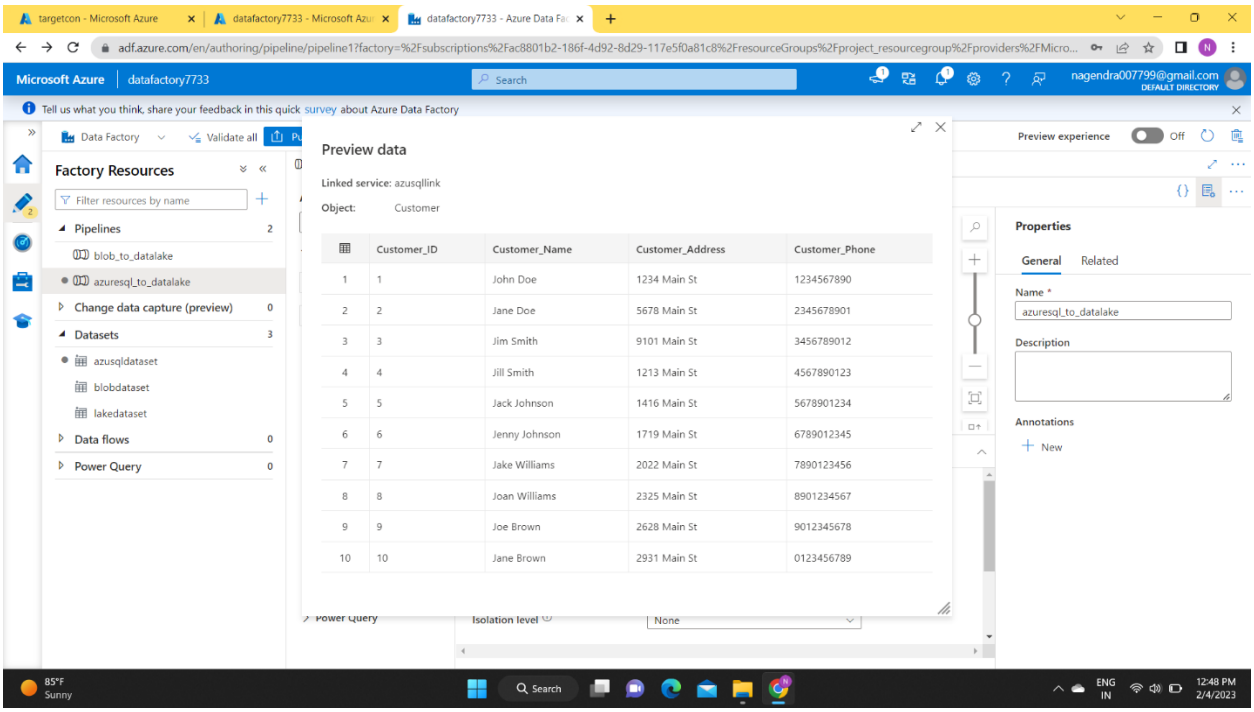
- Here I am changing file name dbo.customer to customer, and I am done the test connection.



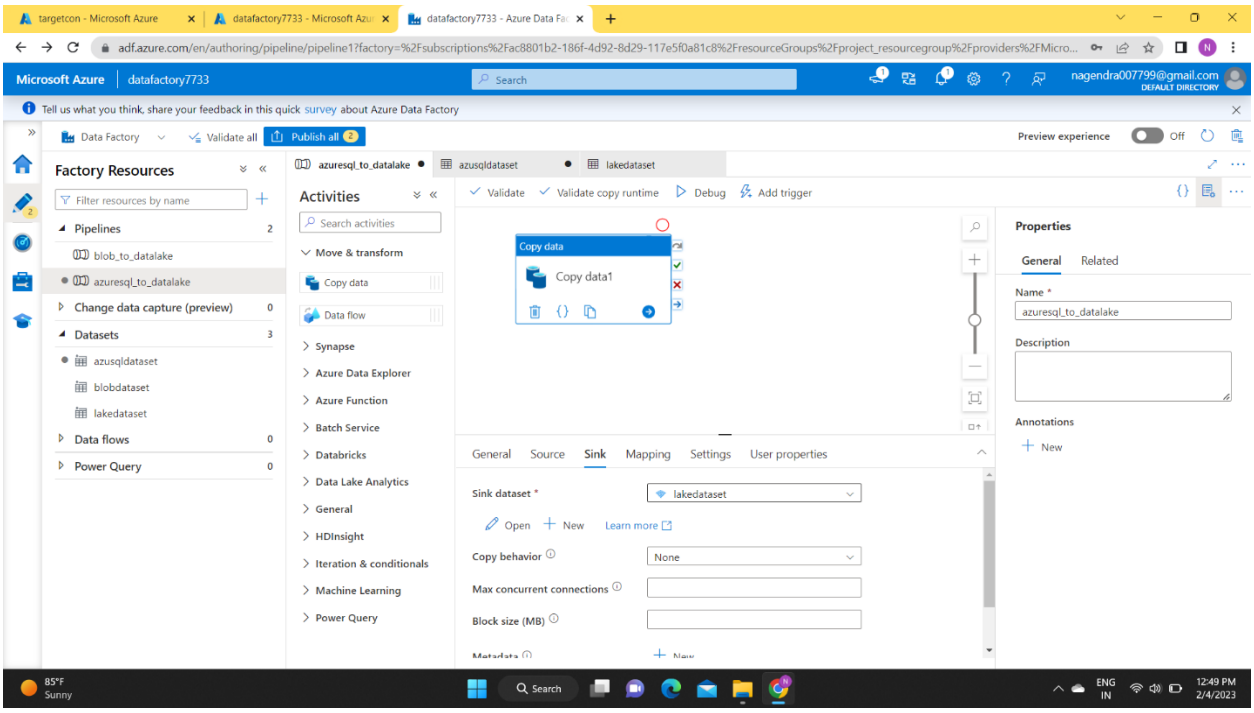
- Now I am going to see preview data



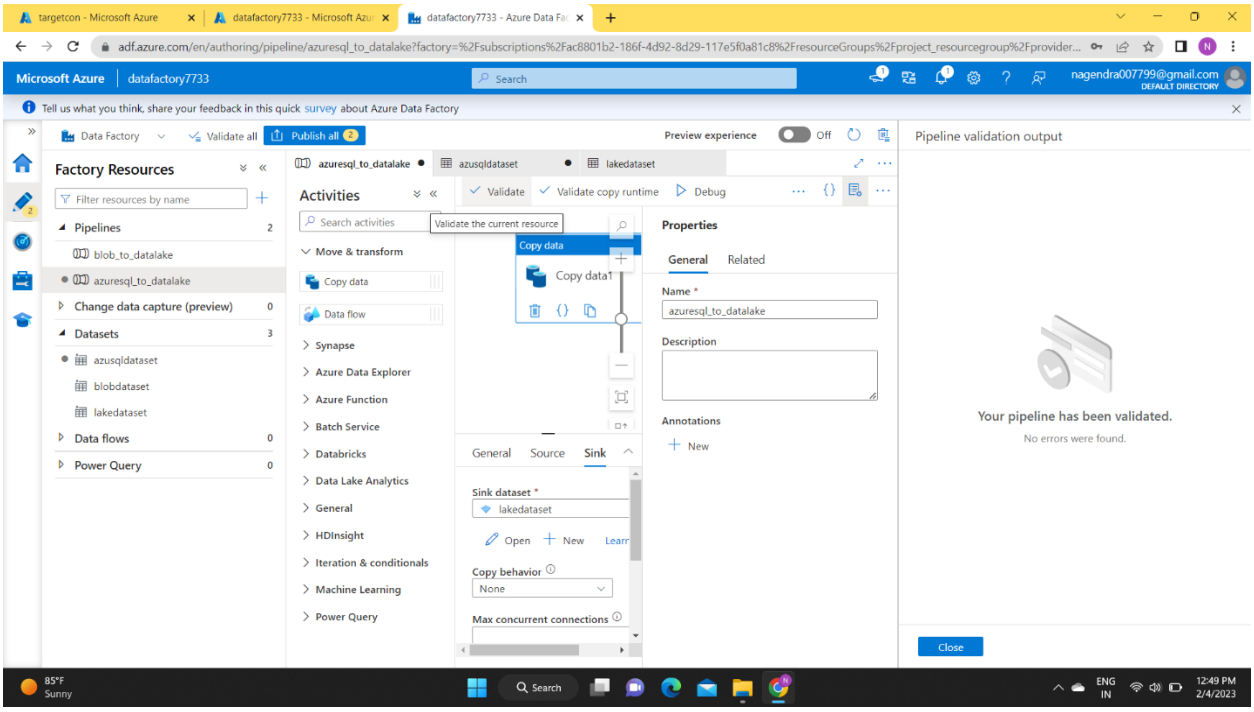
- This is the preview of our customer table data



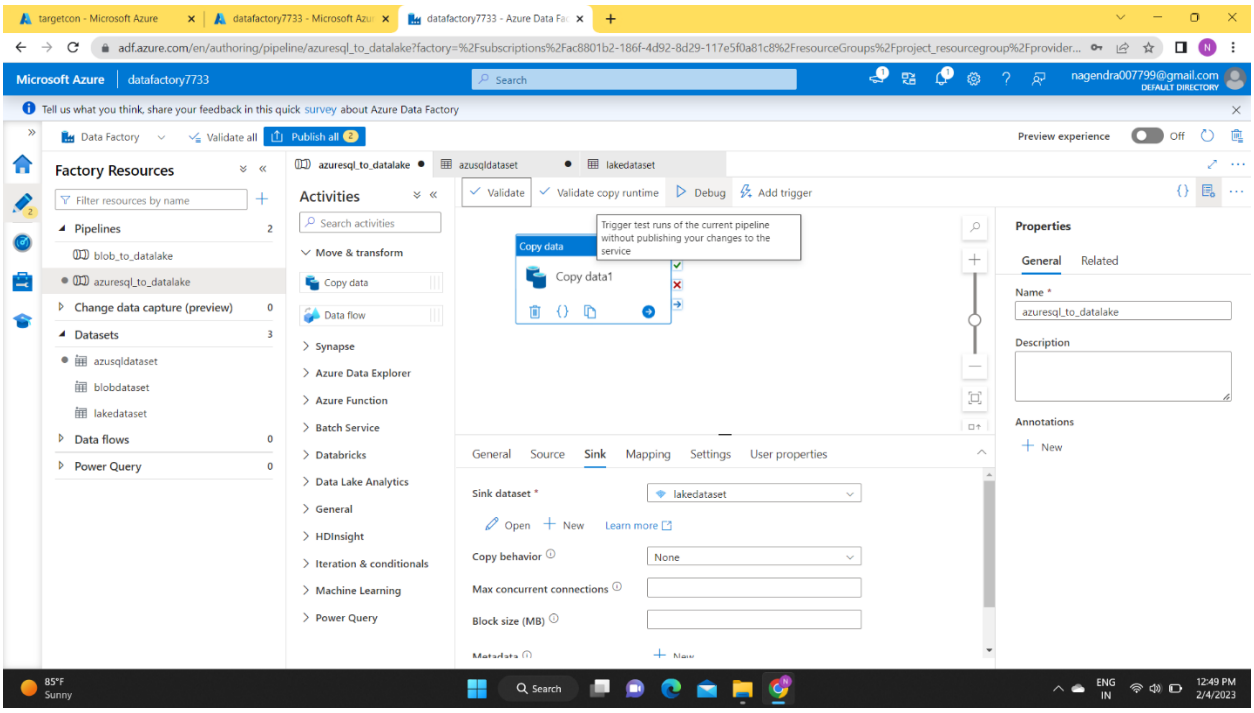
- we have data lake dataset and link service, so I am giving them.



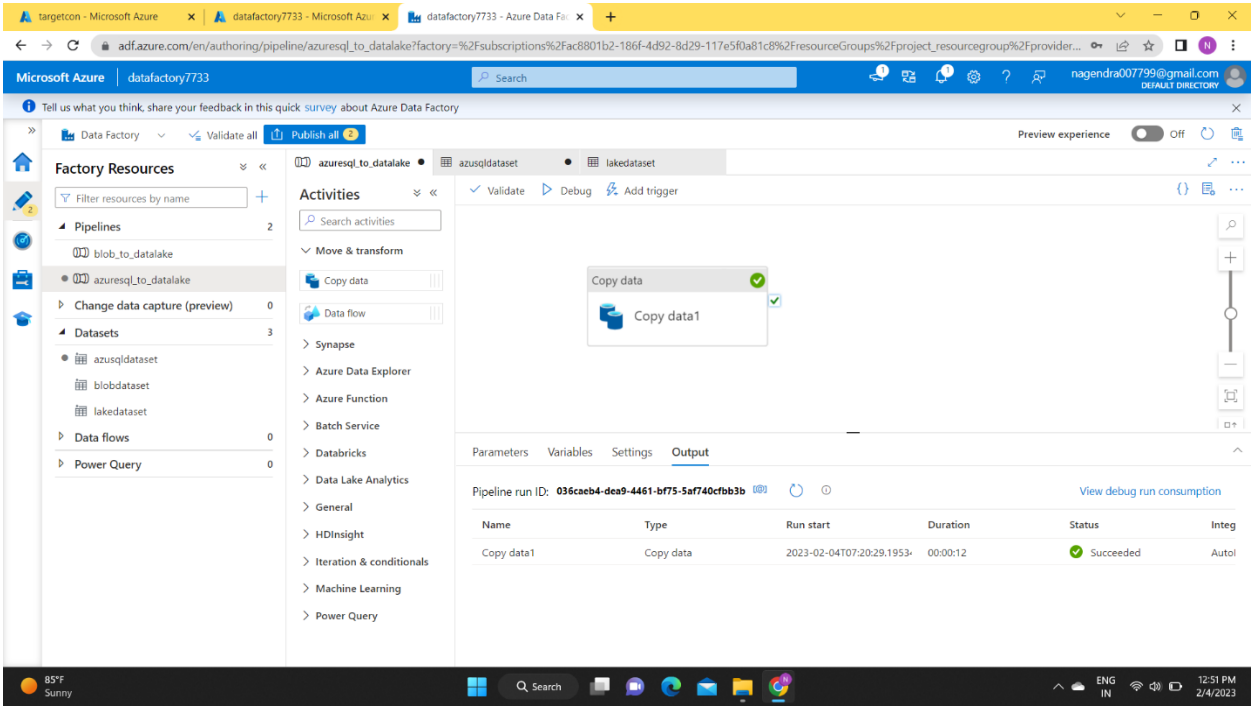
- now I am going to validate our pipeline



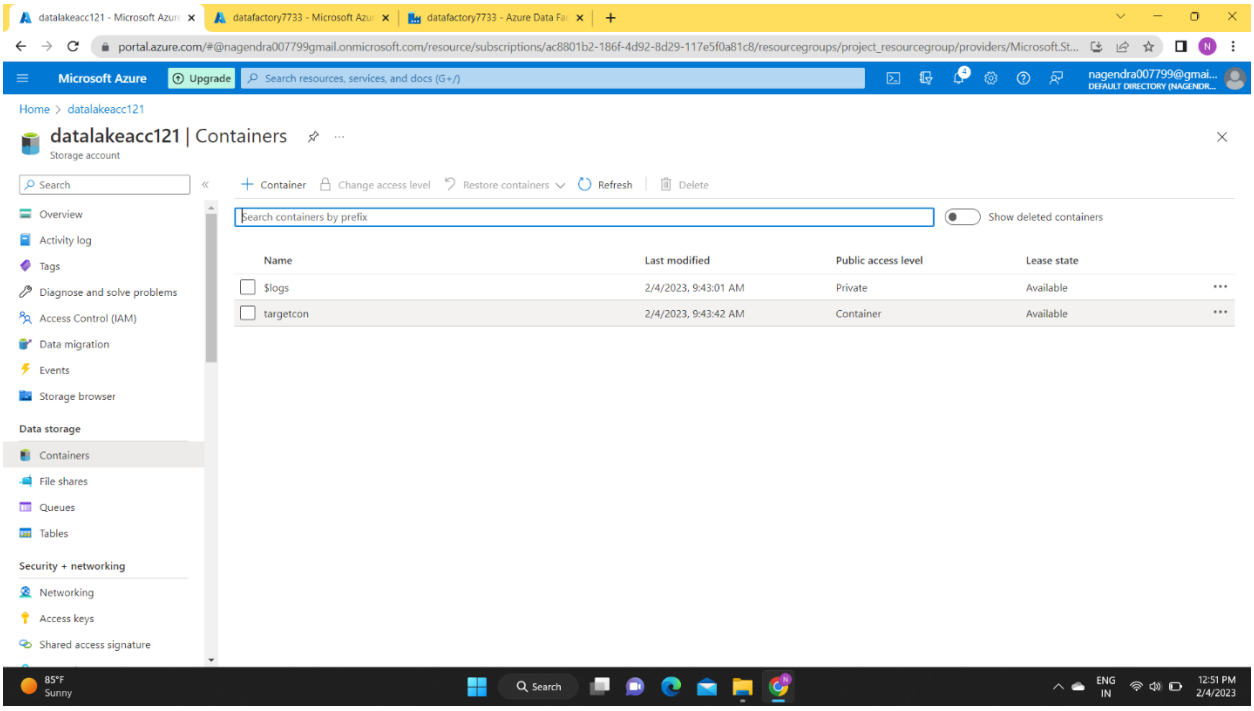
- now I am doing debug



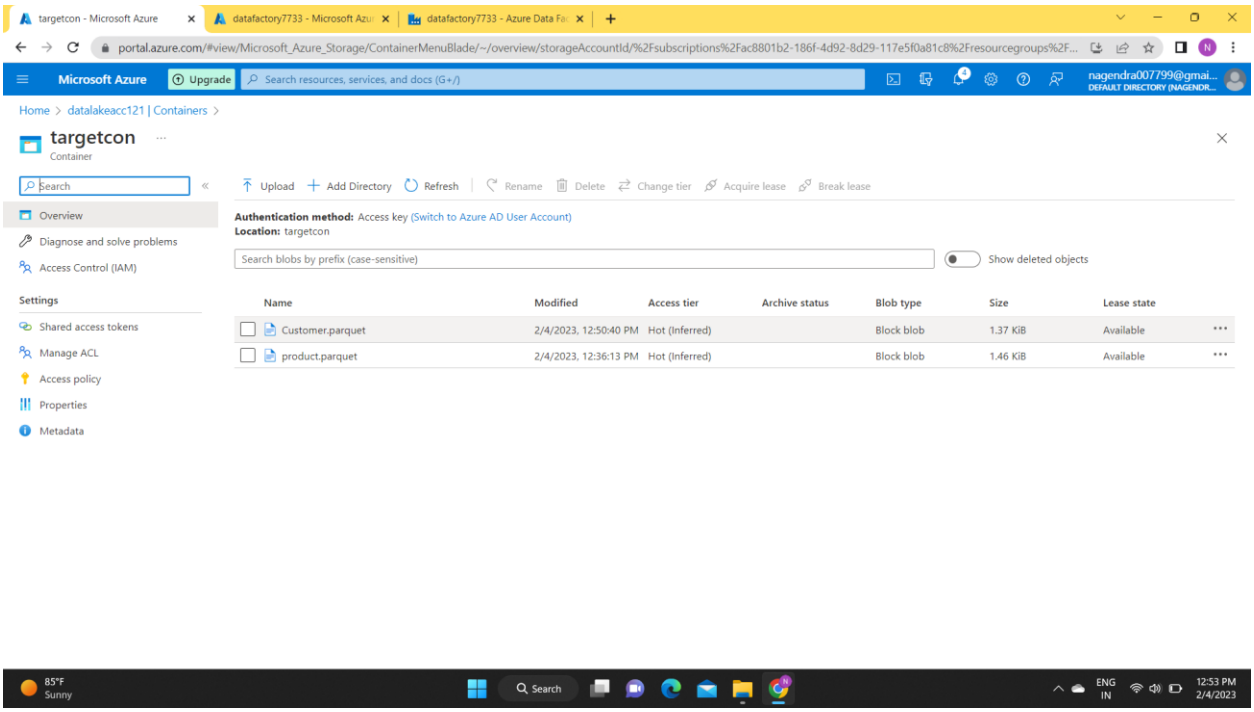
- pipeline is successful.



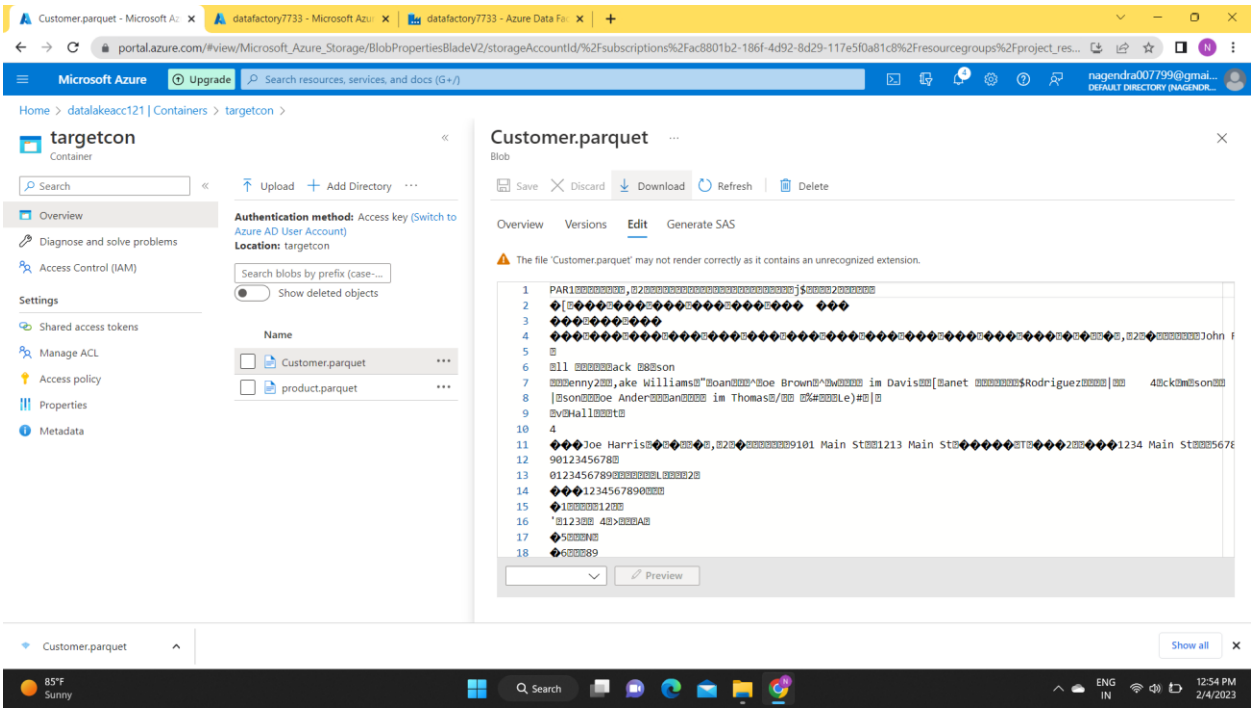
- Now I am checking customer data in target datakake



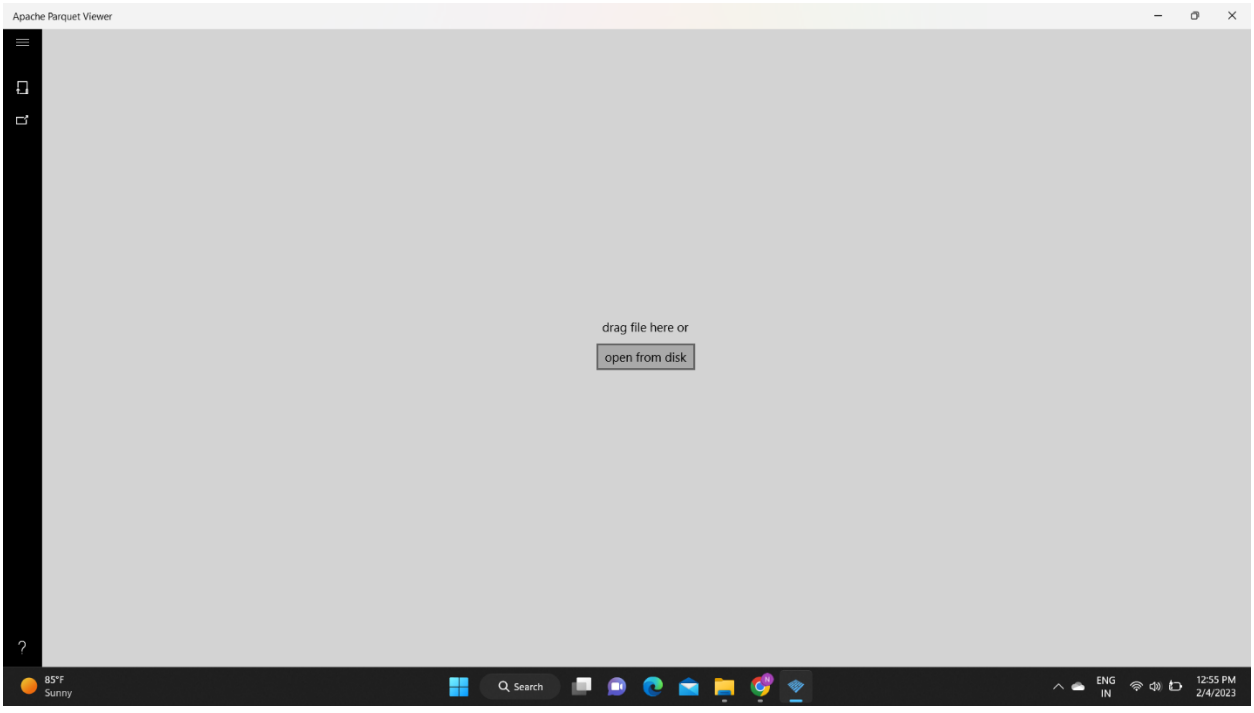
- Here is our customer file in parquet file format.



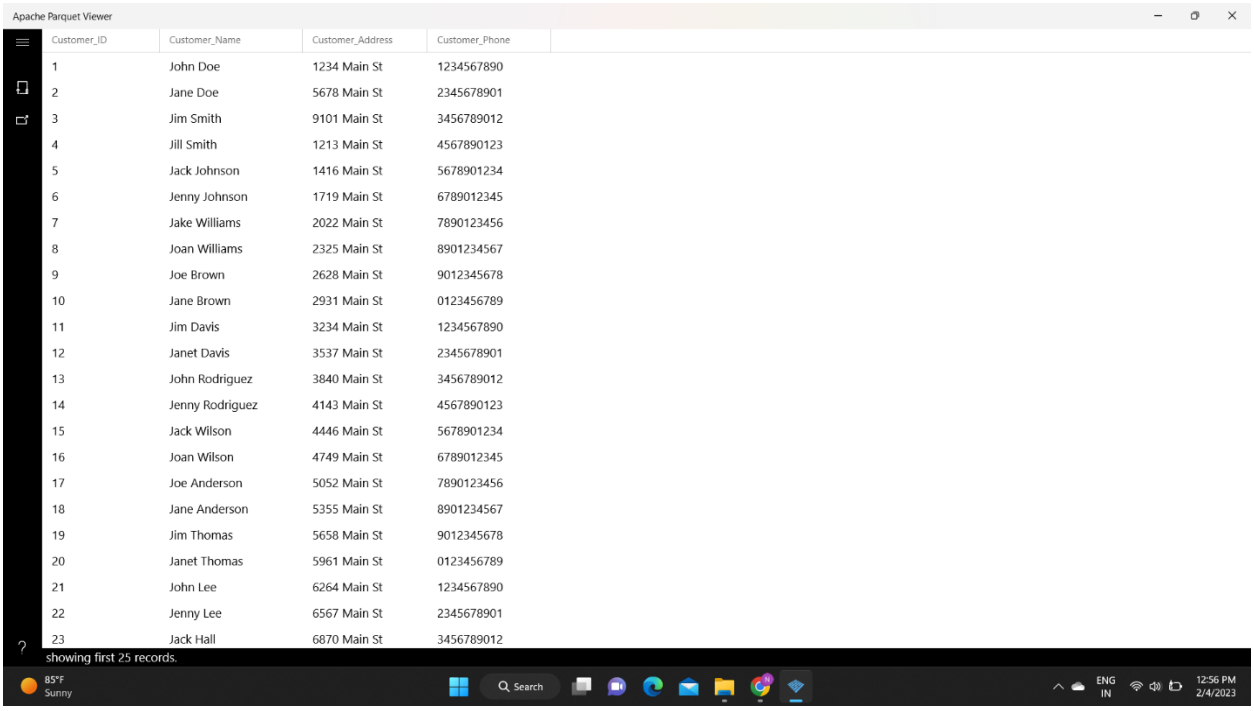
- We cannot understand data so download file.

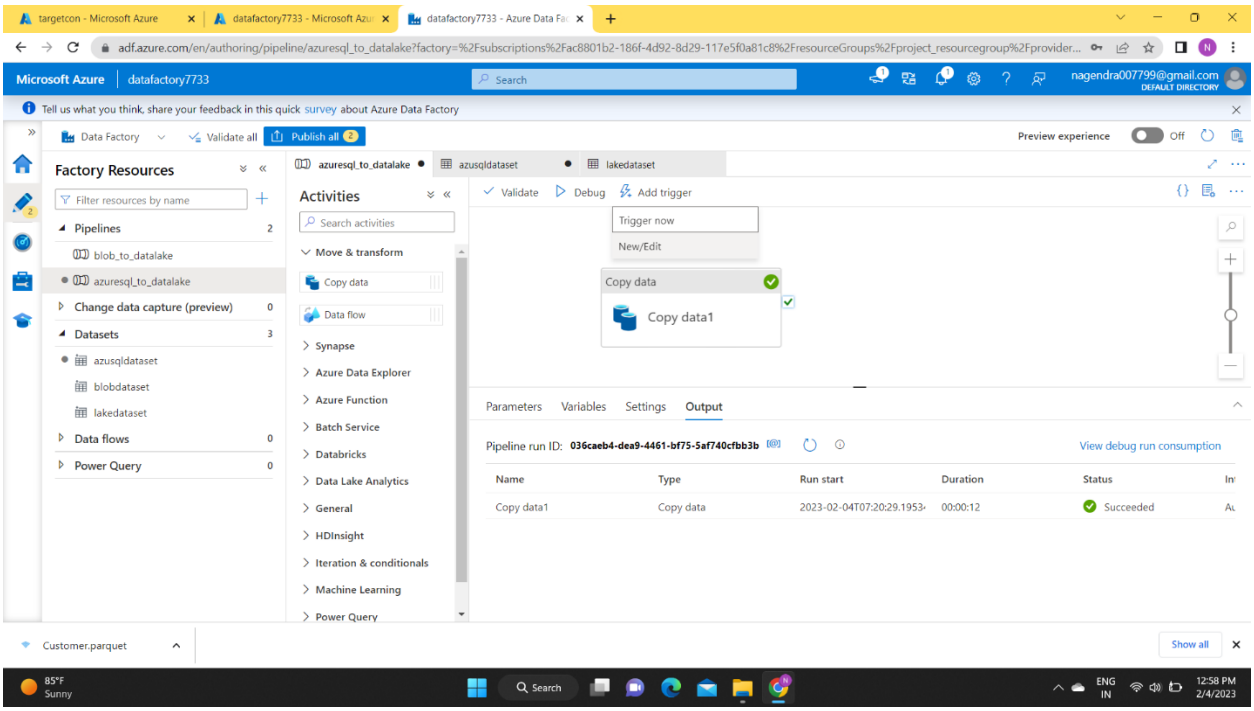


- I am open the app “advanced parquet viewer” and upload your parquet file here.

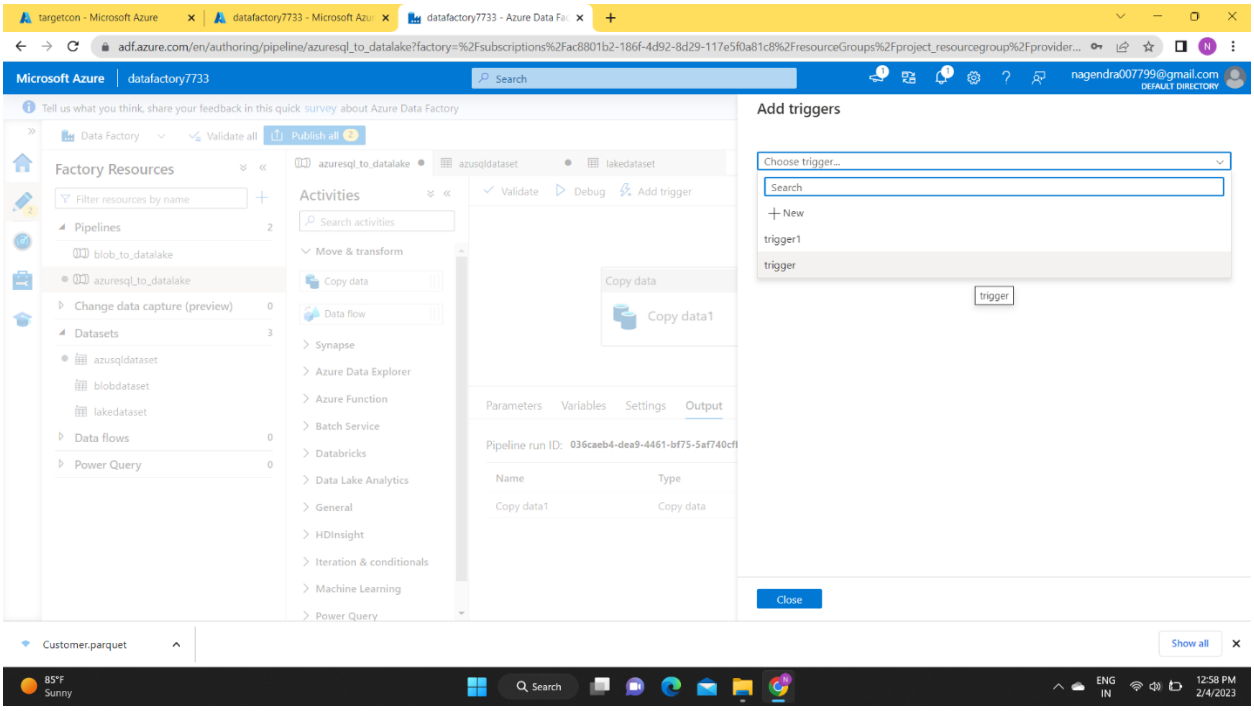


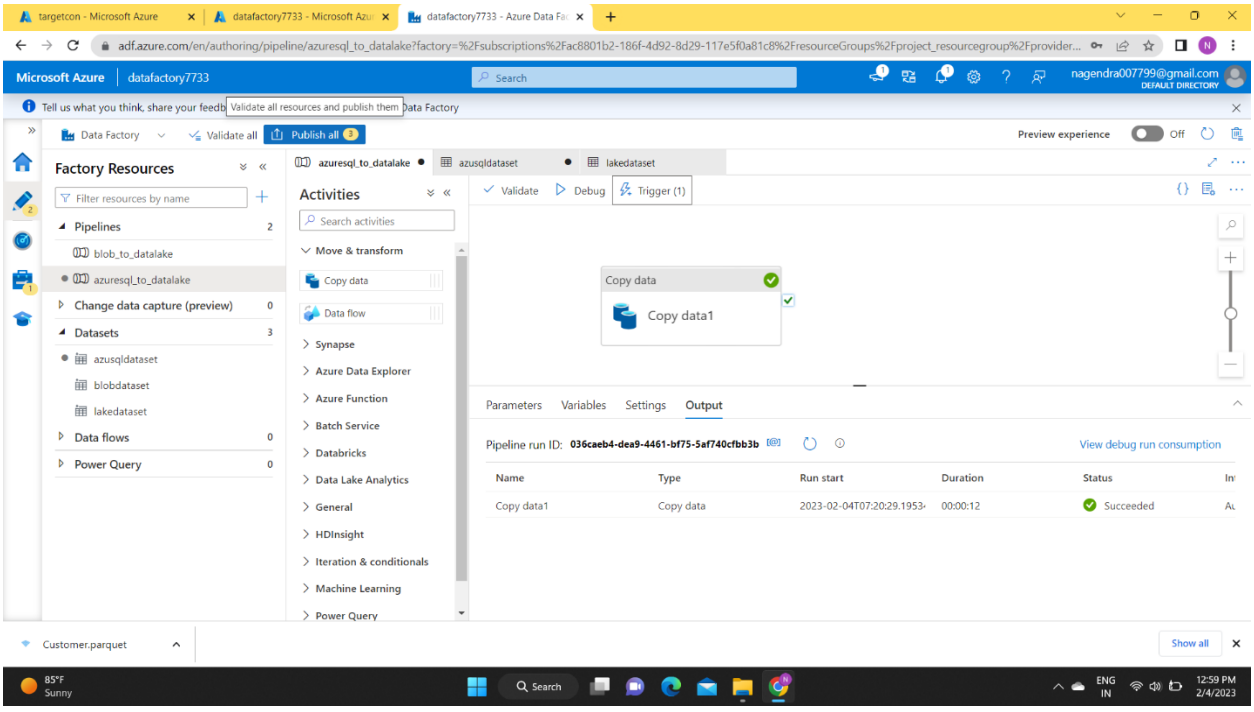
- This is how our customer data looks like.



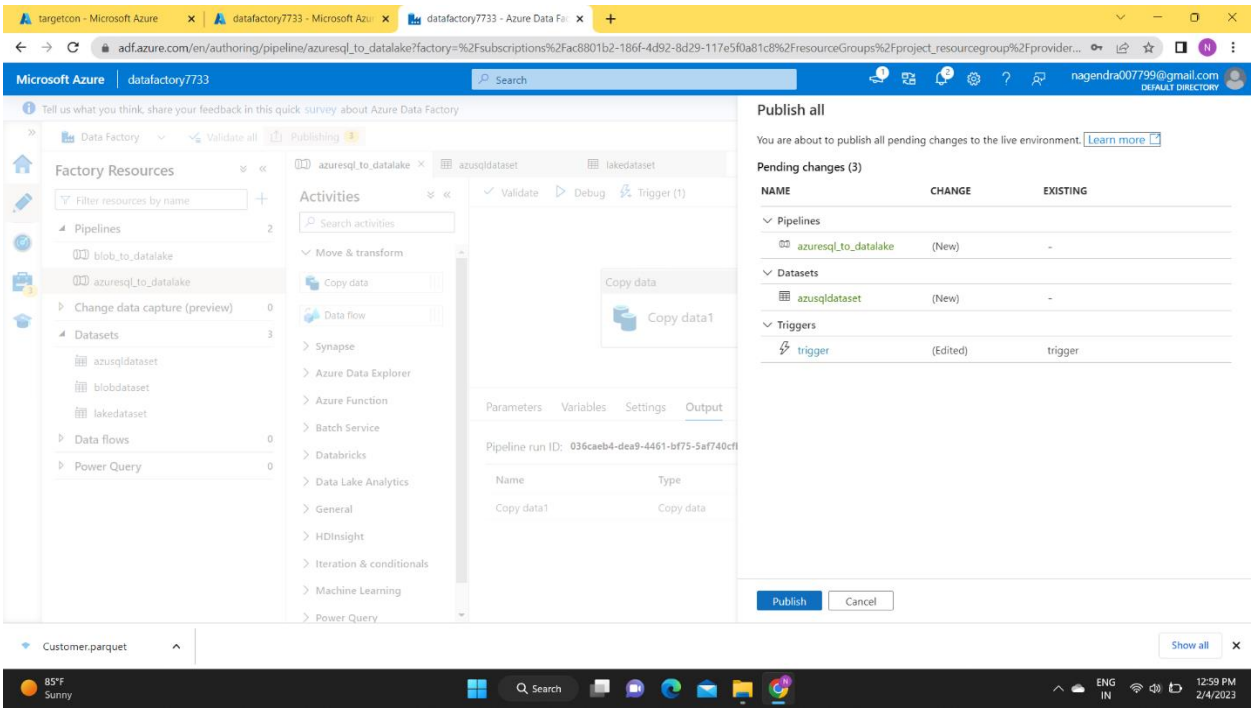


- Now I am adding triggers iam attaching previous trigger which is connected to blob_to_datalake pipeline.

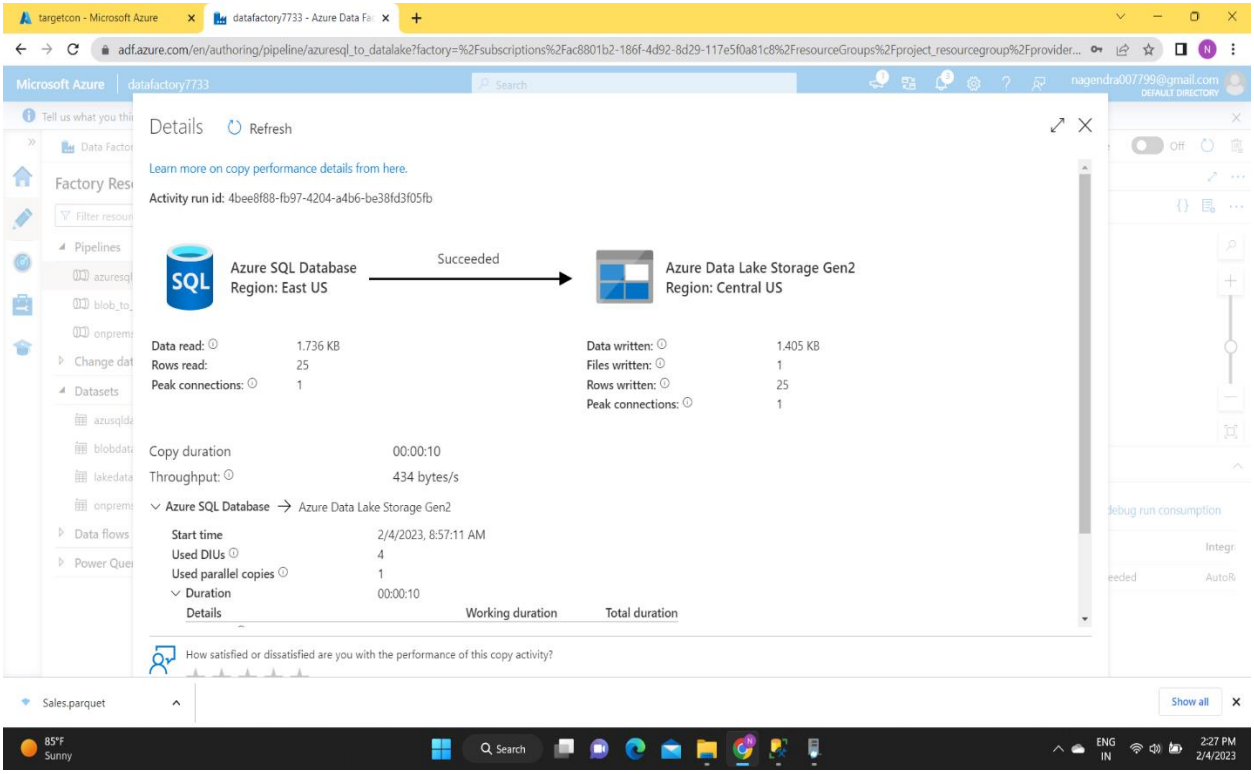




- Now I am publishing the pipeline.

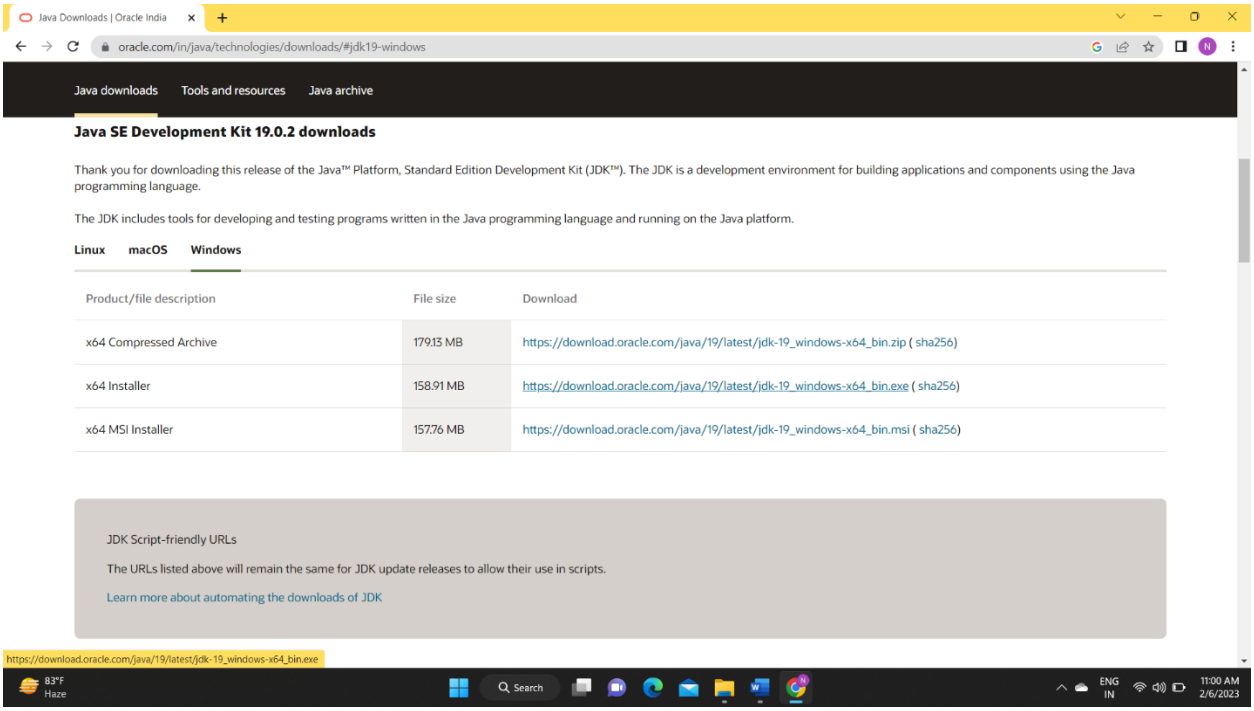


- This is our azure sql to azure datalake pipeline

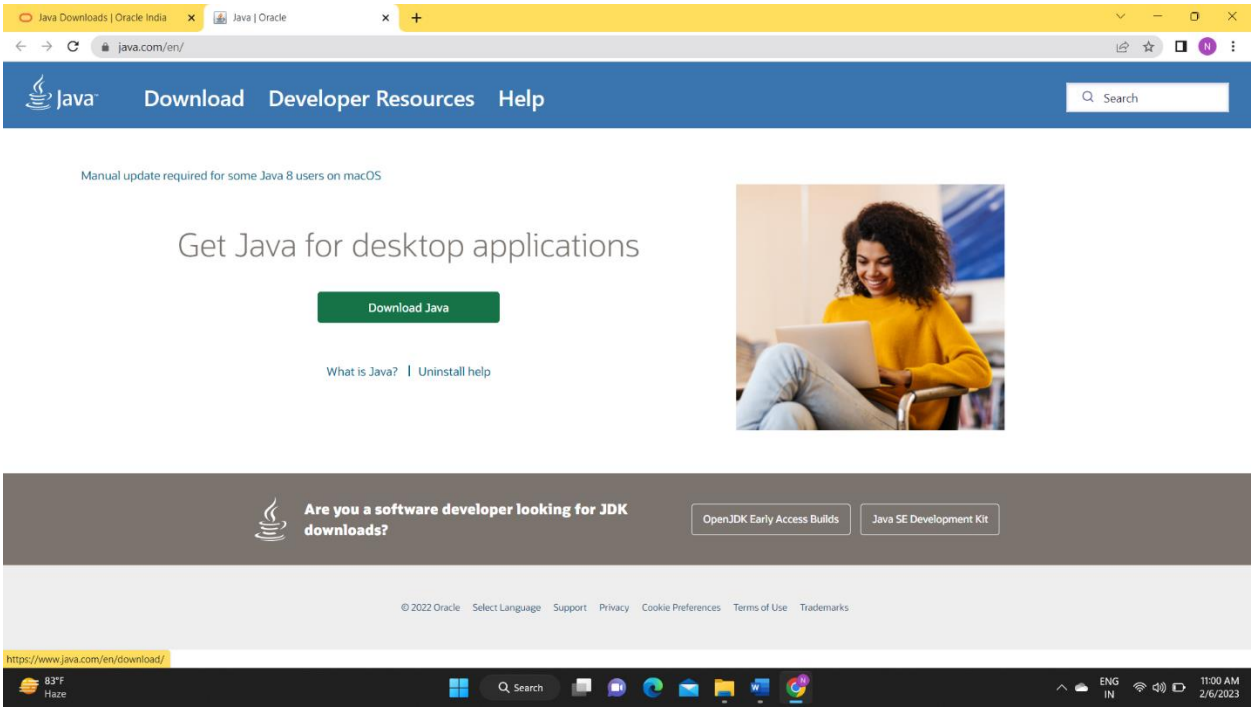


Copy data from on-premises sql server to the azure data lake

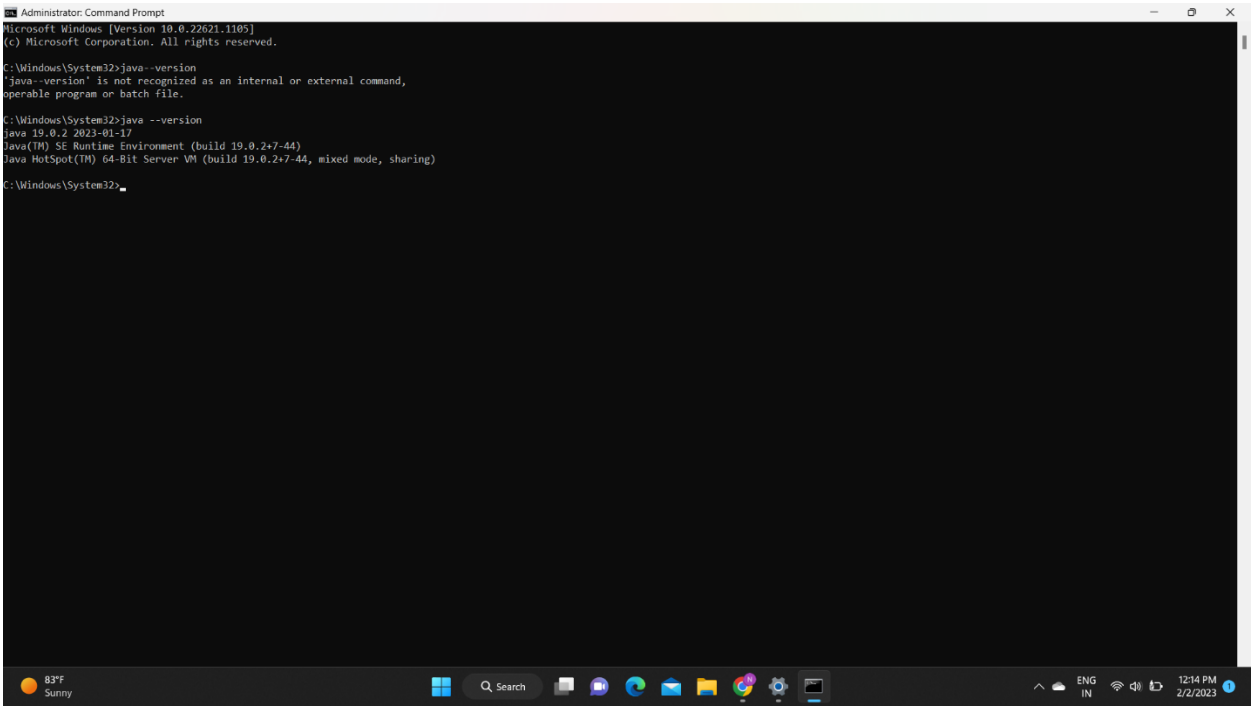
- If you want to fetch on-premises sql server to cloud in parquet file format, java jdk, jre packages must be needed
- Search on google to java jdk download click on first link choose your os and download X64 installer
- Link to download jdk https://www.oracle.com/in/java/technologies/downloads/



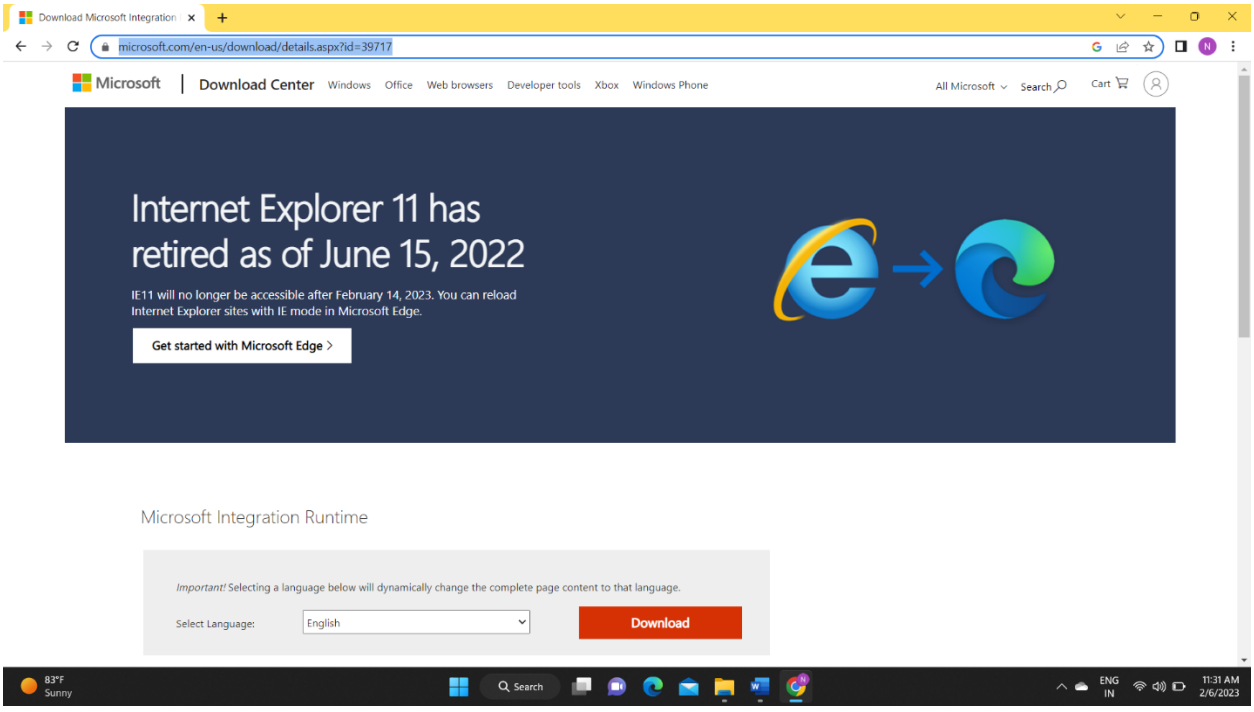
- After jdk download search on java jre download and download jre.
- Link to download jre https://www.java.com/en/



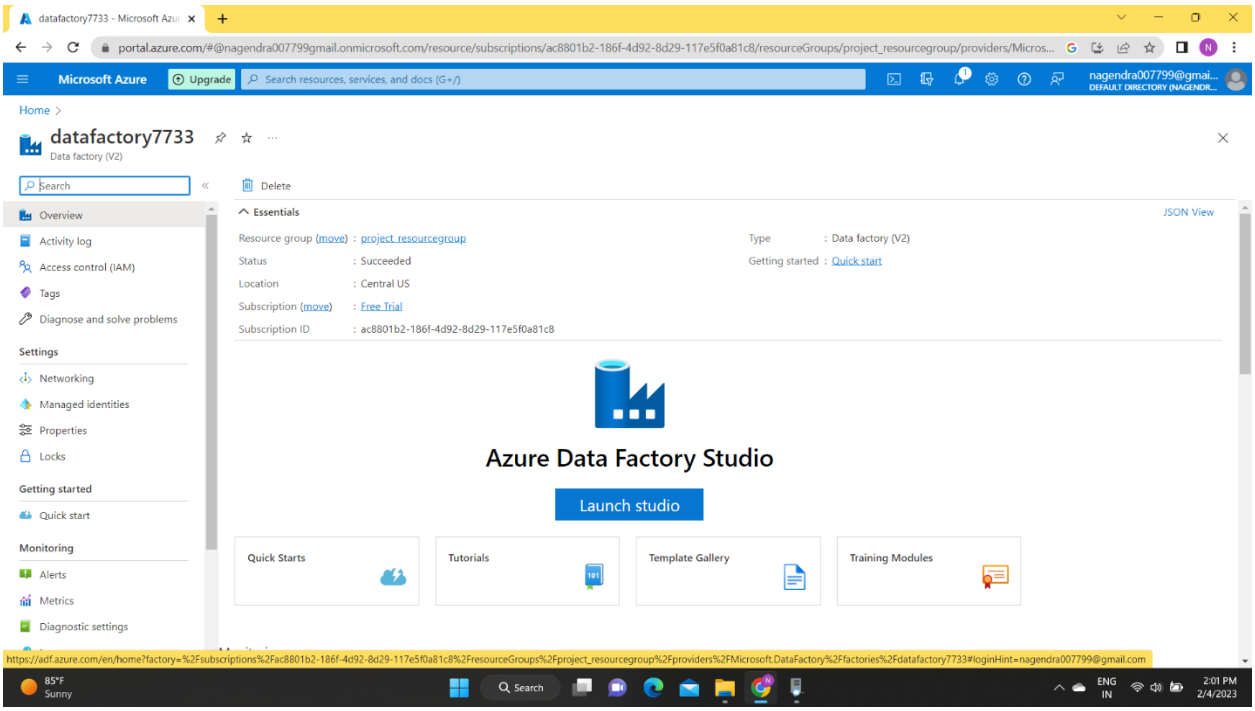
- You can check whether java packages are installed or not in window terminal



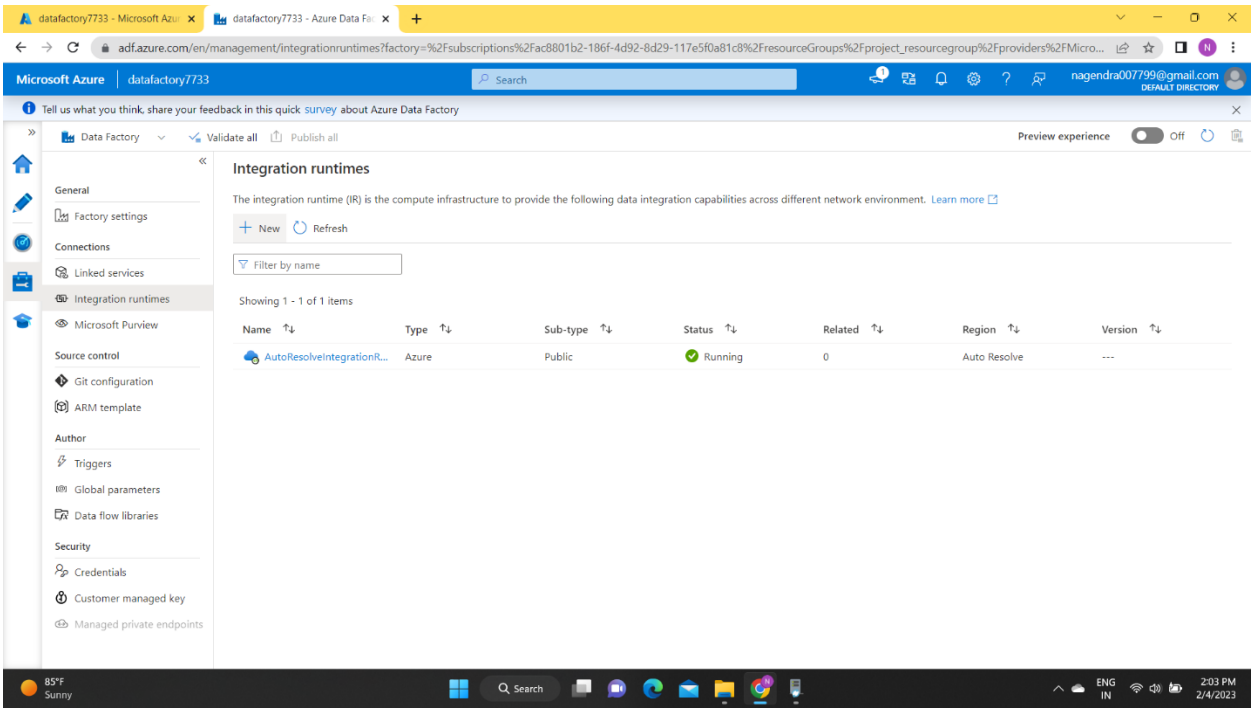
- There is a software Microsoft integration runtime we have to install in our on-premises server
- So it connect to azure integration runtime
- Link to download Microsoft integration runtime
  https://www.microsoft.com/en-us/download/details.aspx?id=39717



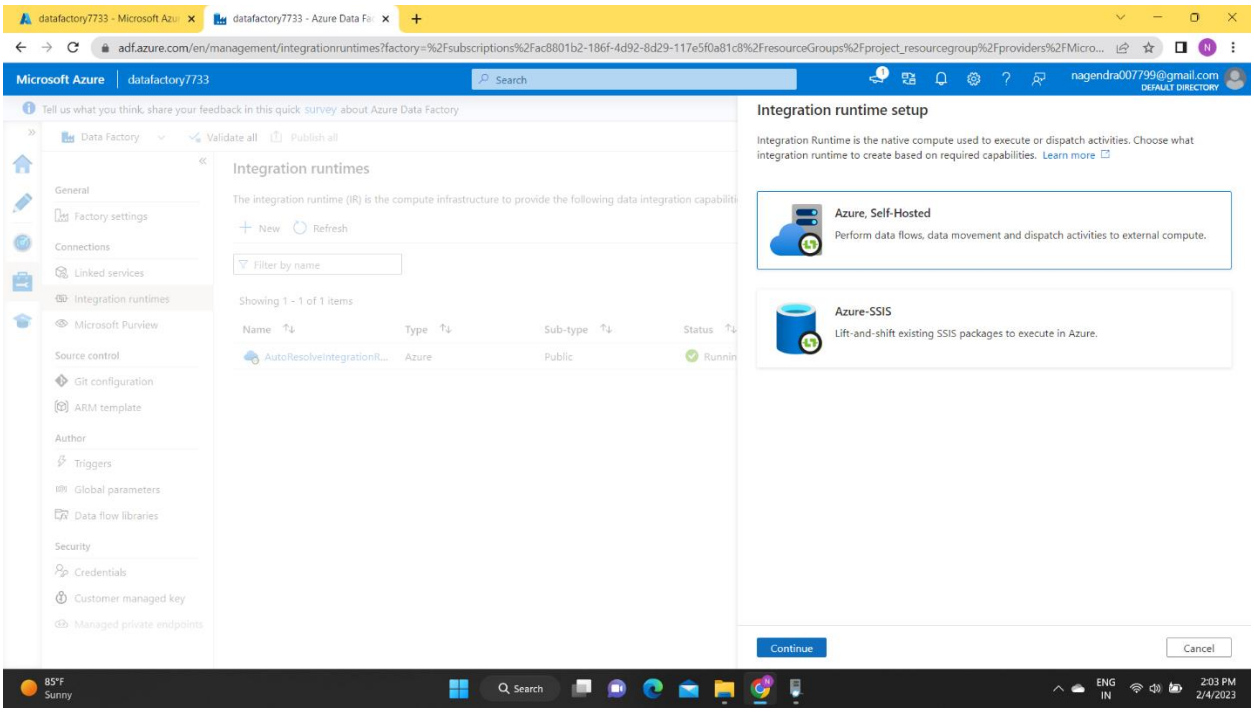
- Launch the azure data factory

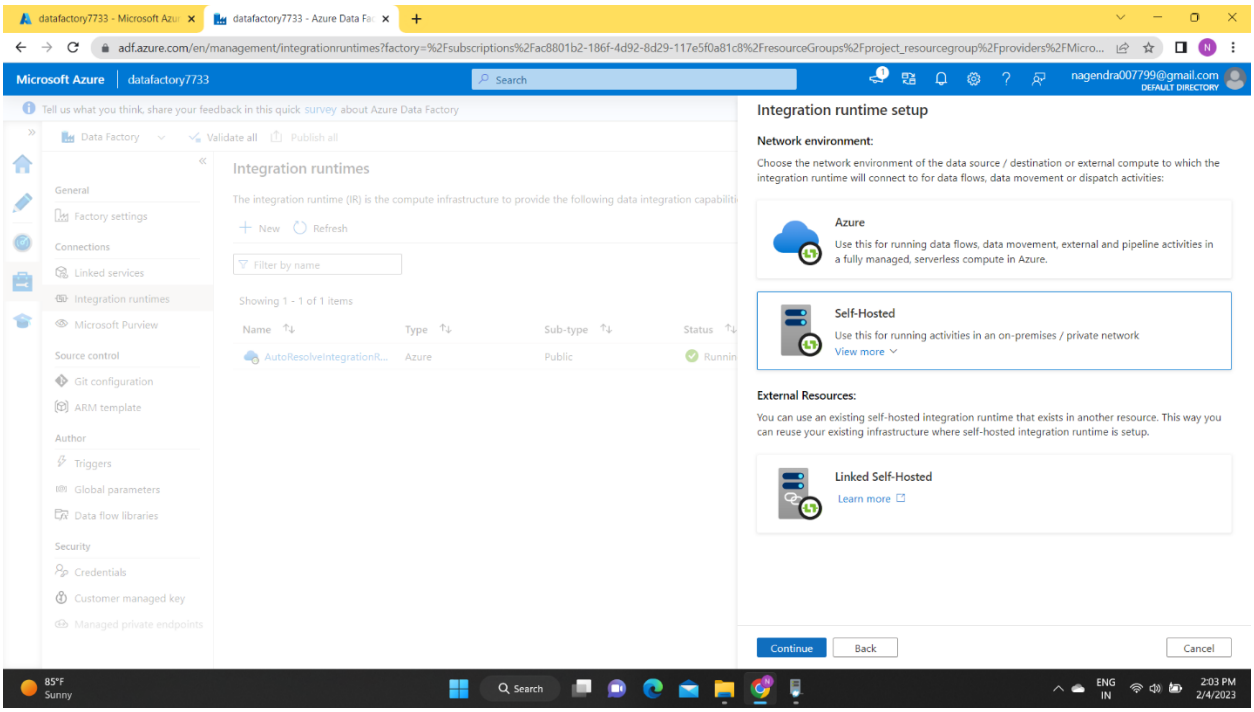


- Go to integration runtime and one “auto resolve integration runtime” is running which is commonly used in azure integration.
- Click on +new symbol.

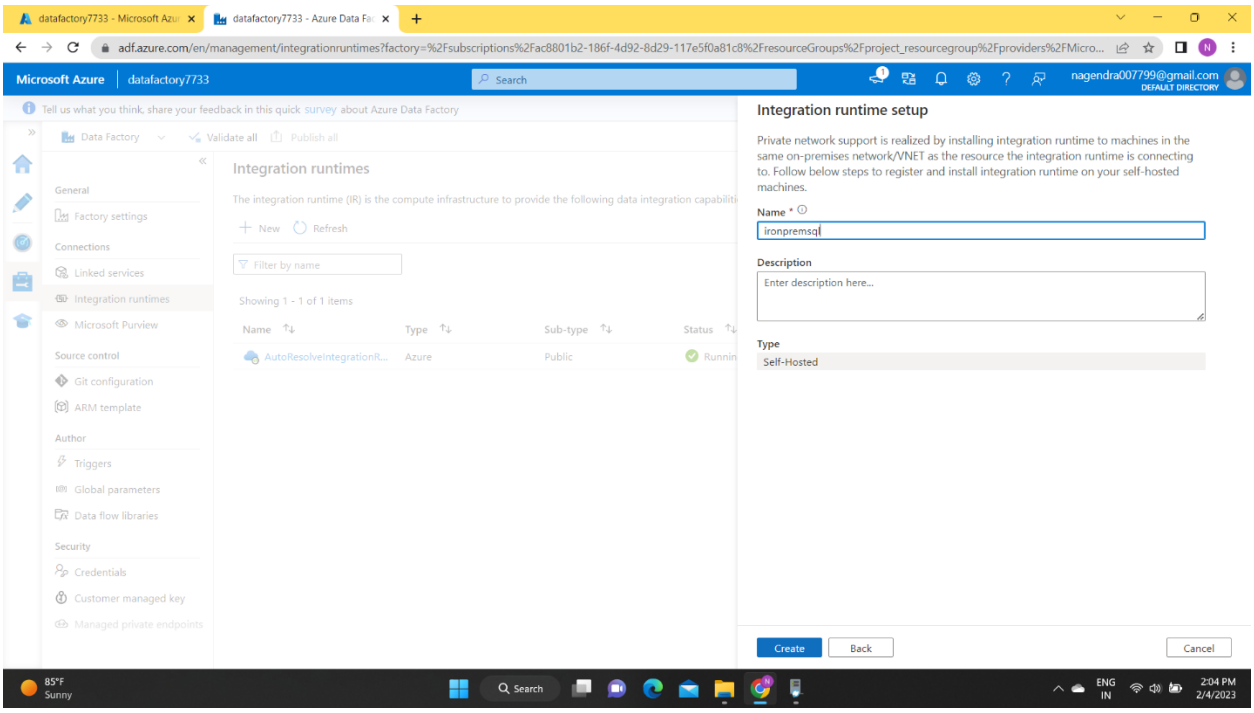


- Select azure self hosted runtime.

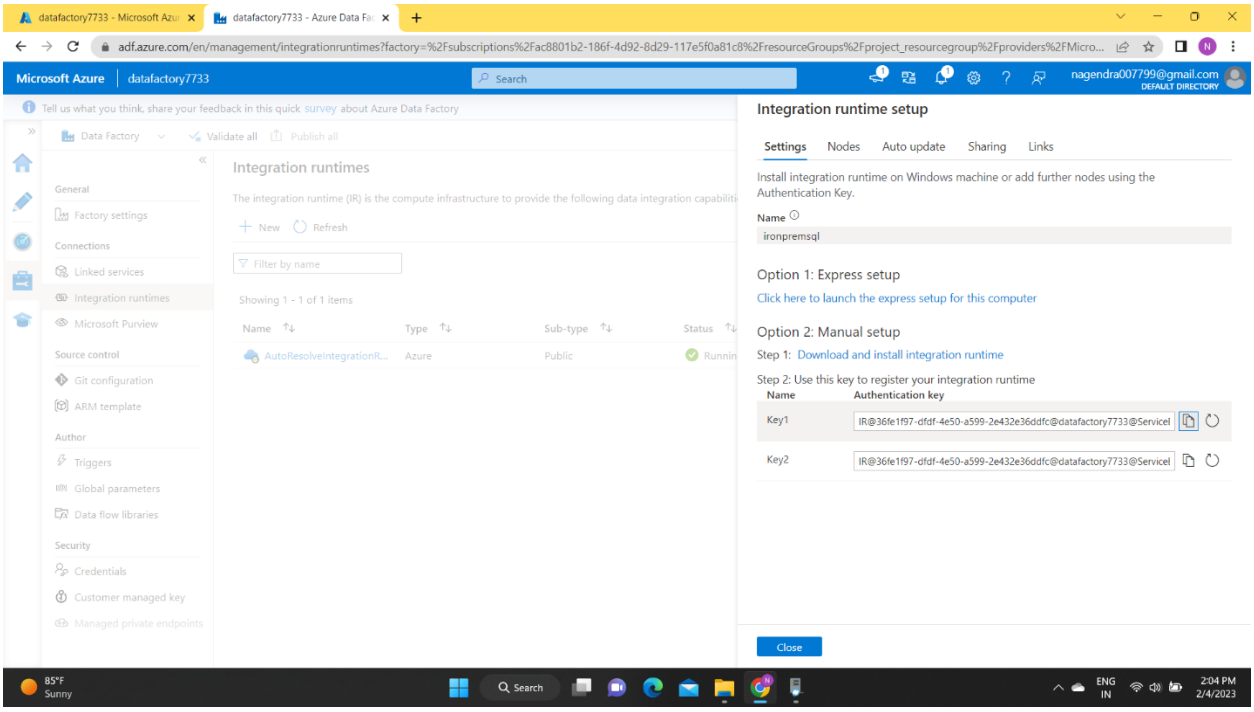




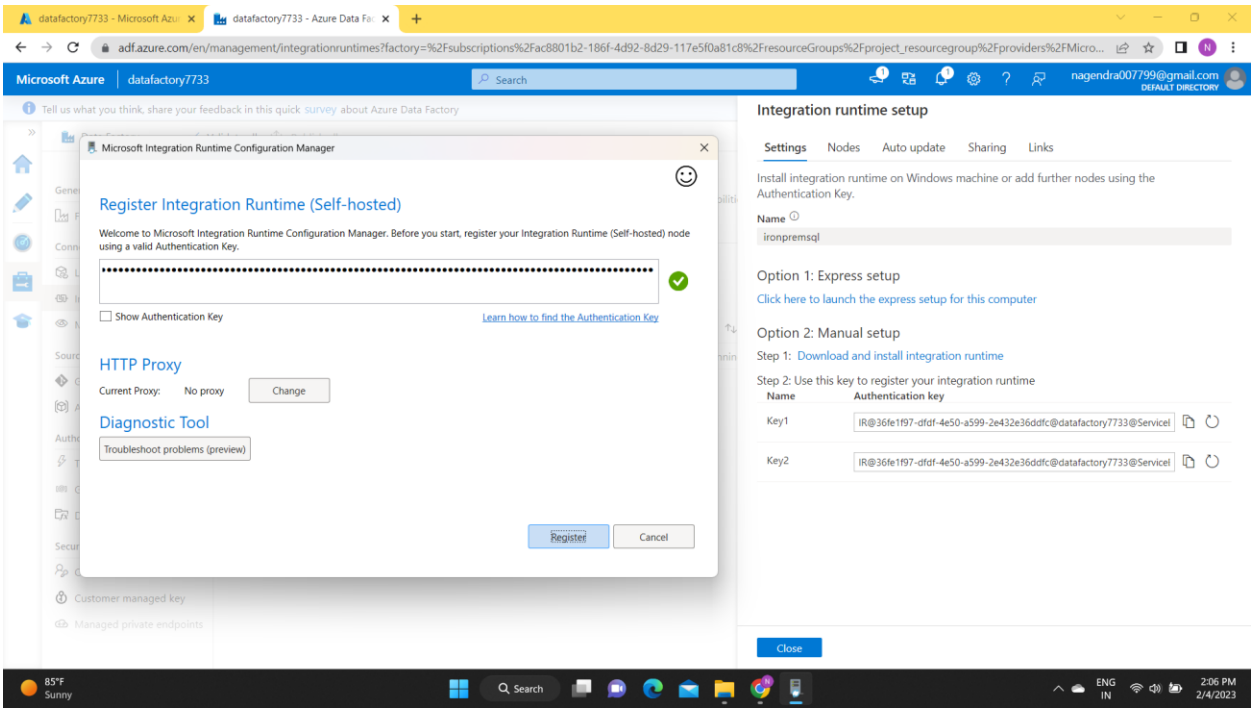
- Give the name to the integration runtime “ironpremsql”.



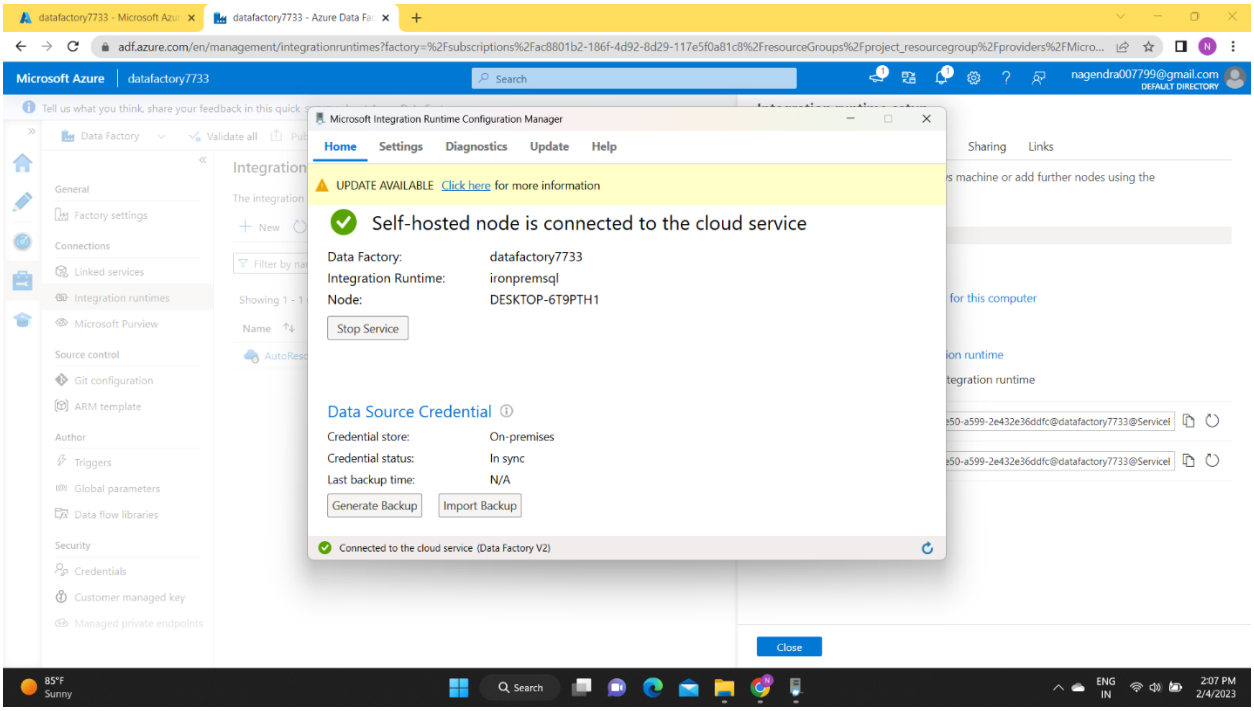
- Copy any link from these two.



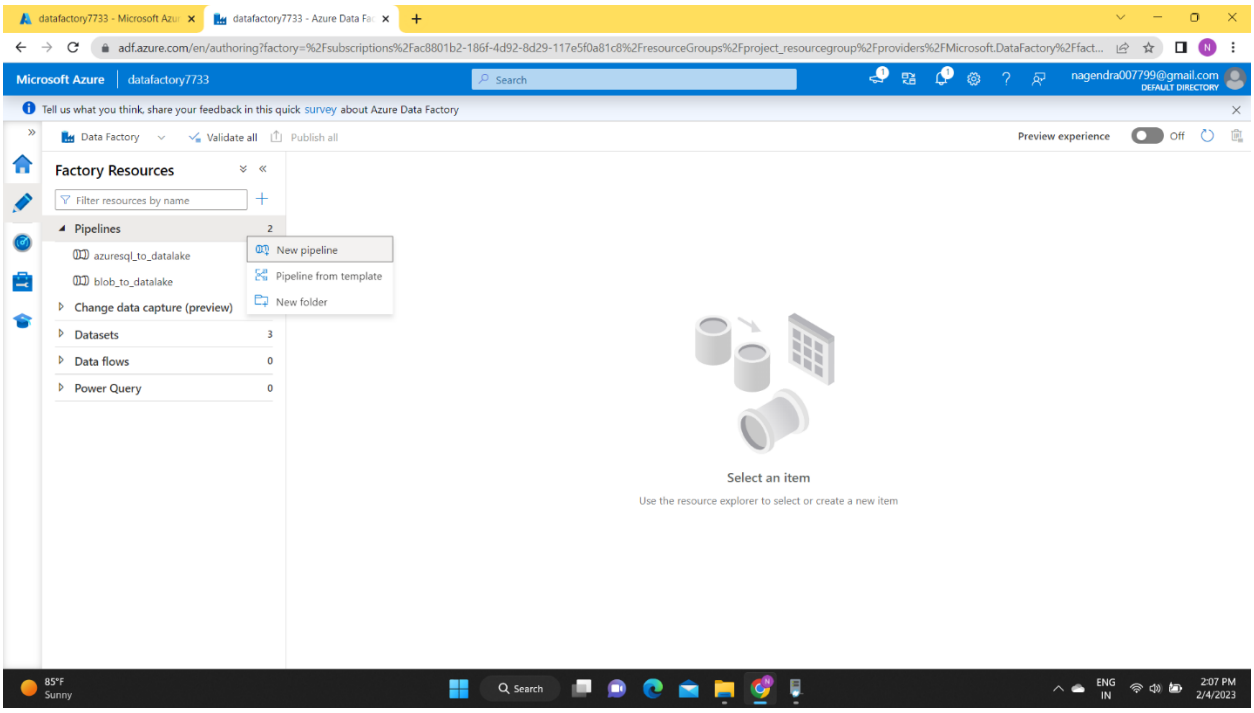
- Open windows integration runtime and paste the link in that box.



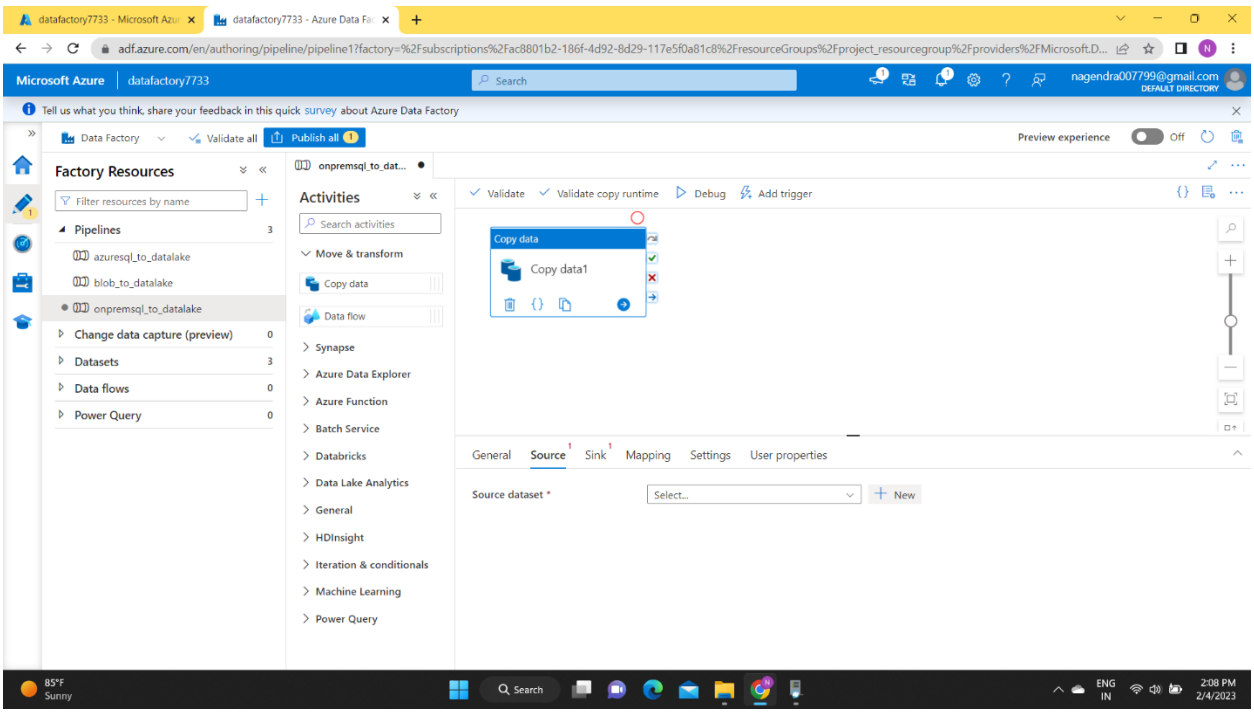
- Now our local windows server connected to azure cloud.



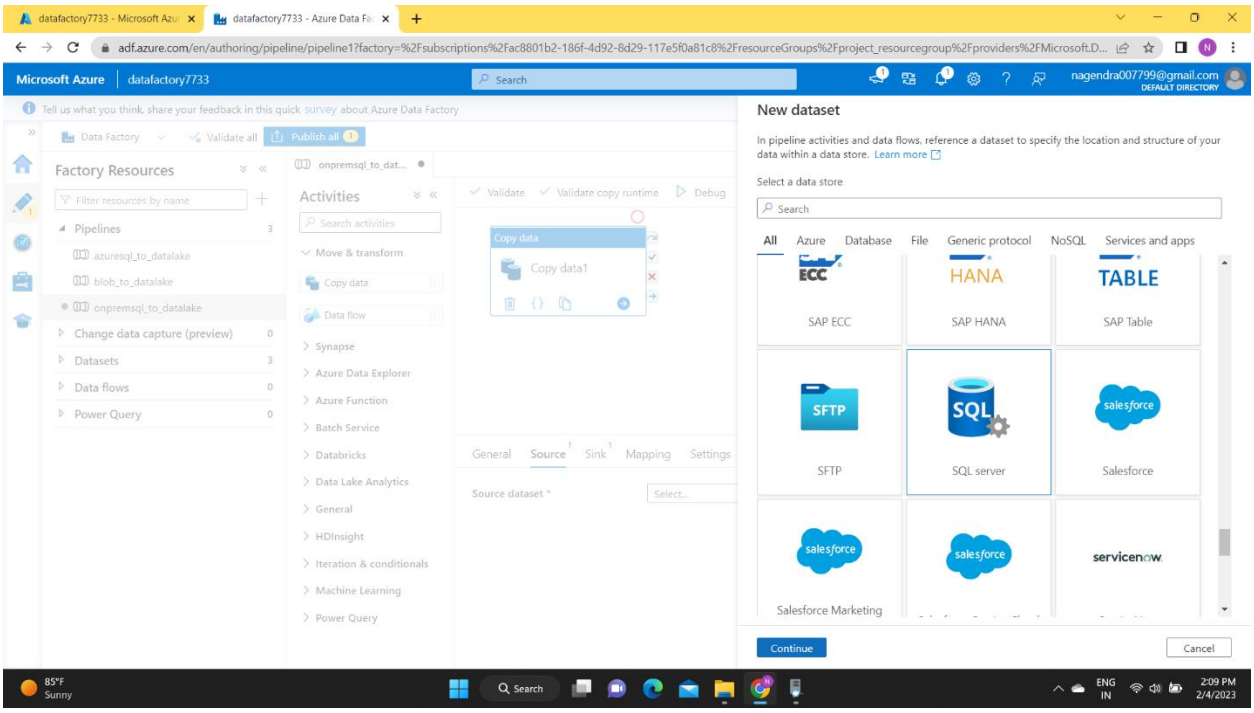
- Open the adf and create new pipe line “onpremsql_to_azuresqldb.



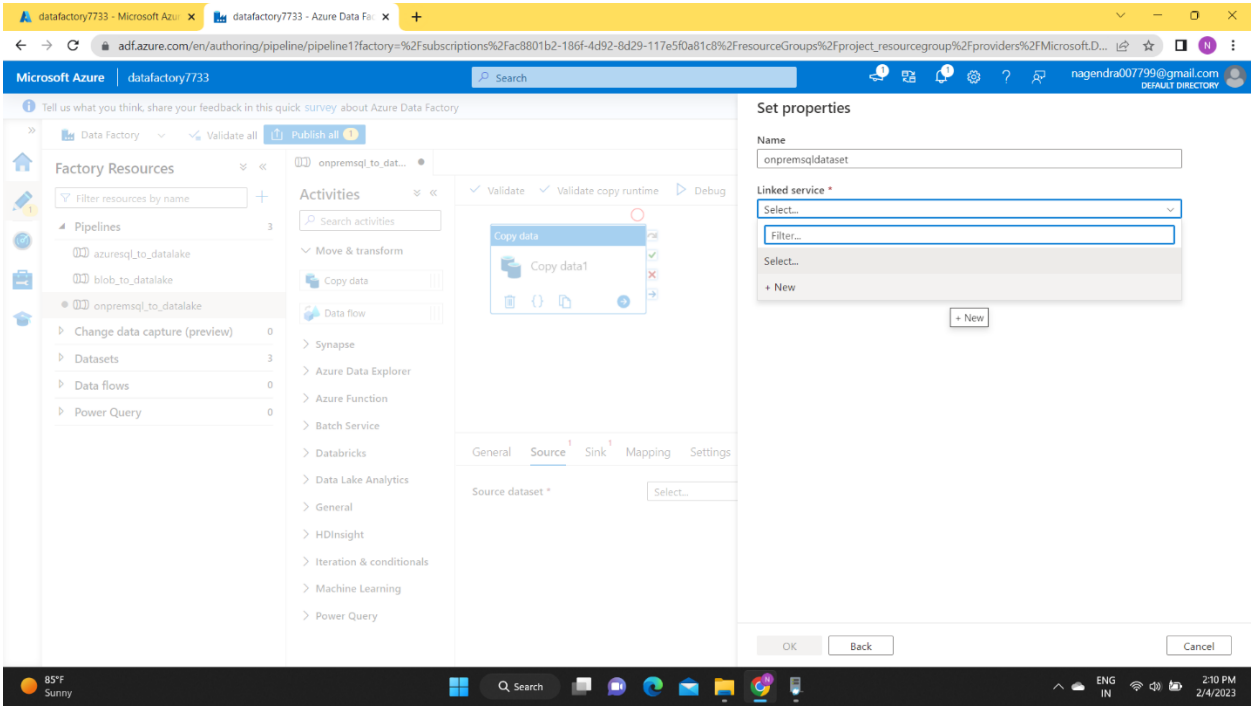
- Drag the copy activity from move and transform.



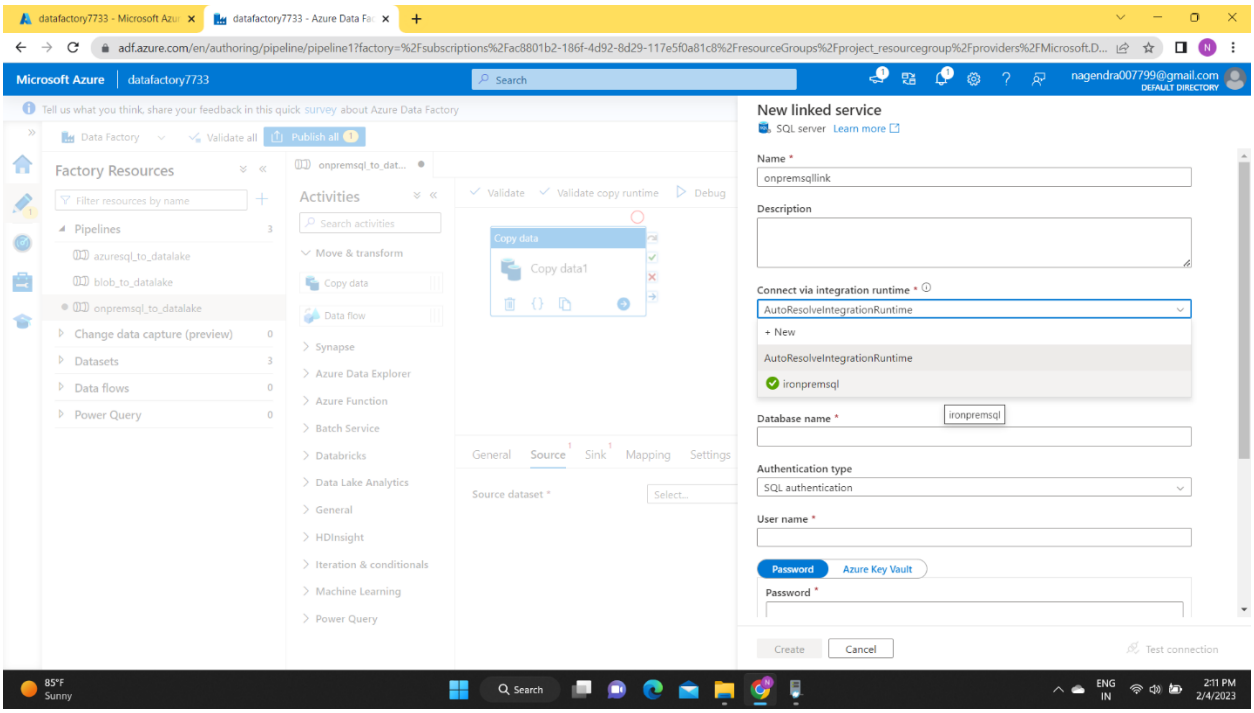
- We don’t need to create sql server we are using on-premises sql server details .
- Select sql server for the source dataset.

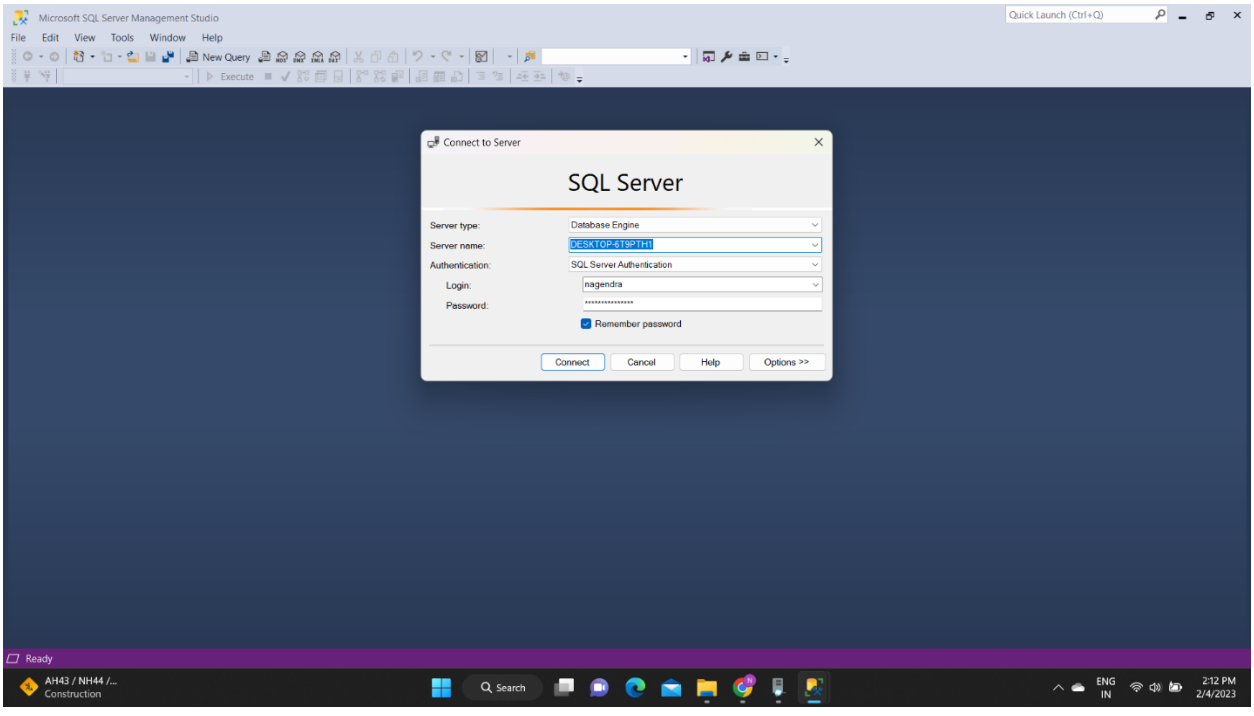


- Click on create new link service to on-premises sql server.
- Sql server dataset name is “onpremsqldataset”

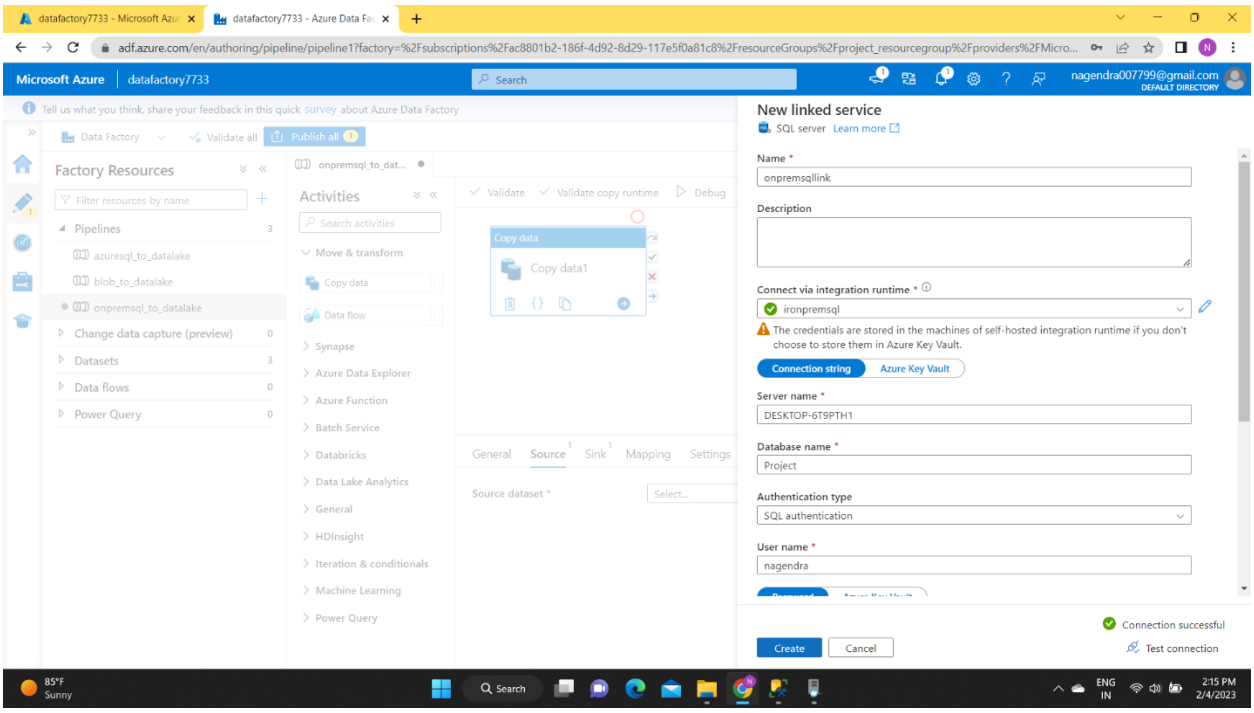


- Our SQL server link service name is “onpremsqllink”
- In connect via integration runtime box select our “ironpremsql”
- You can fill remaining details by using below pic

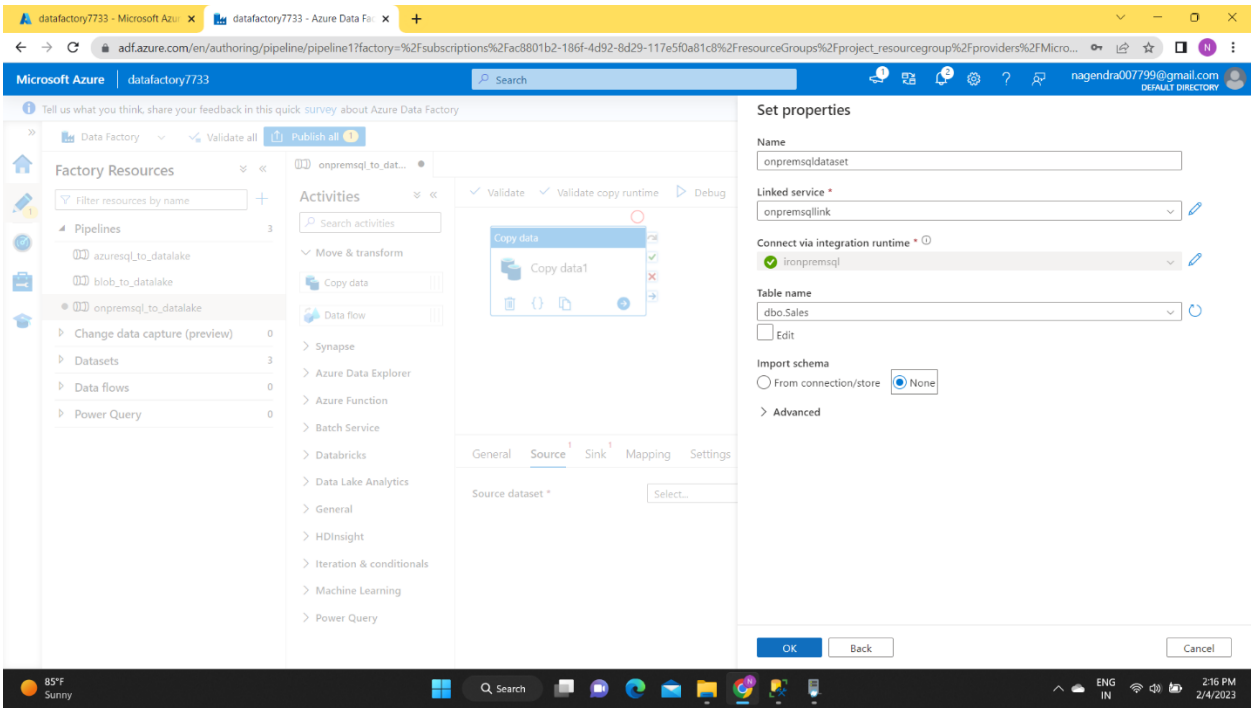




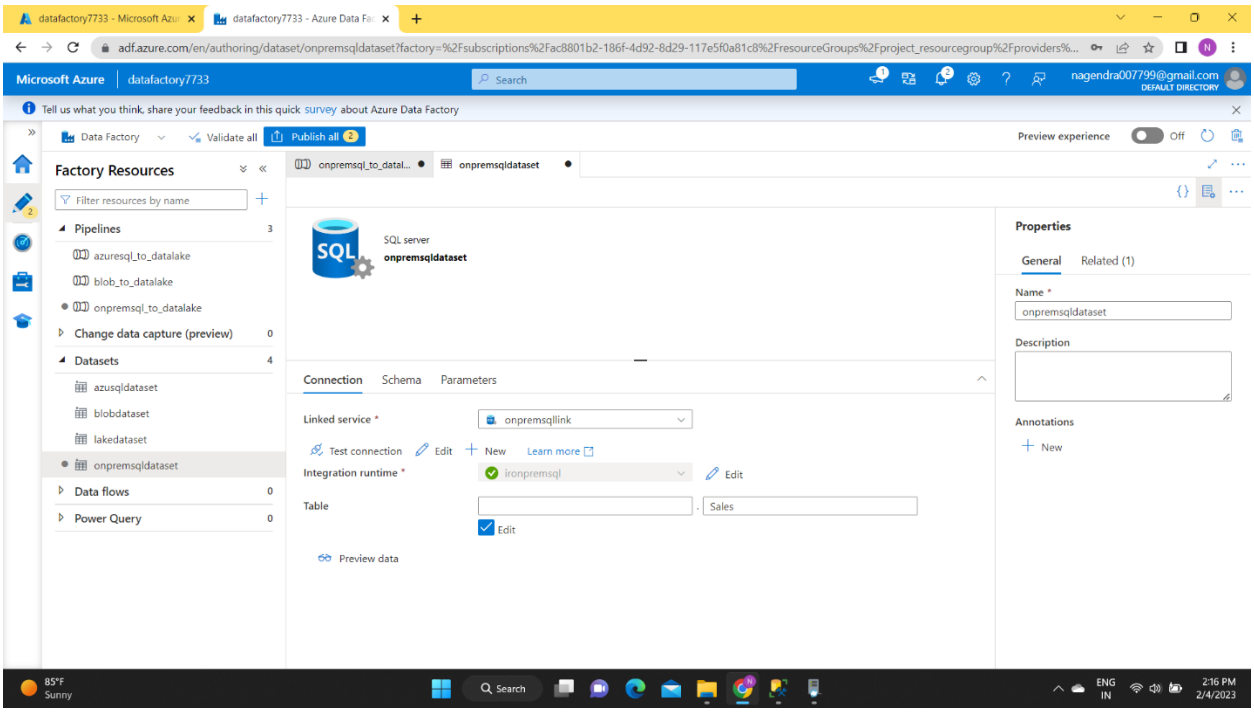
- Do the test connection.



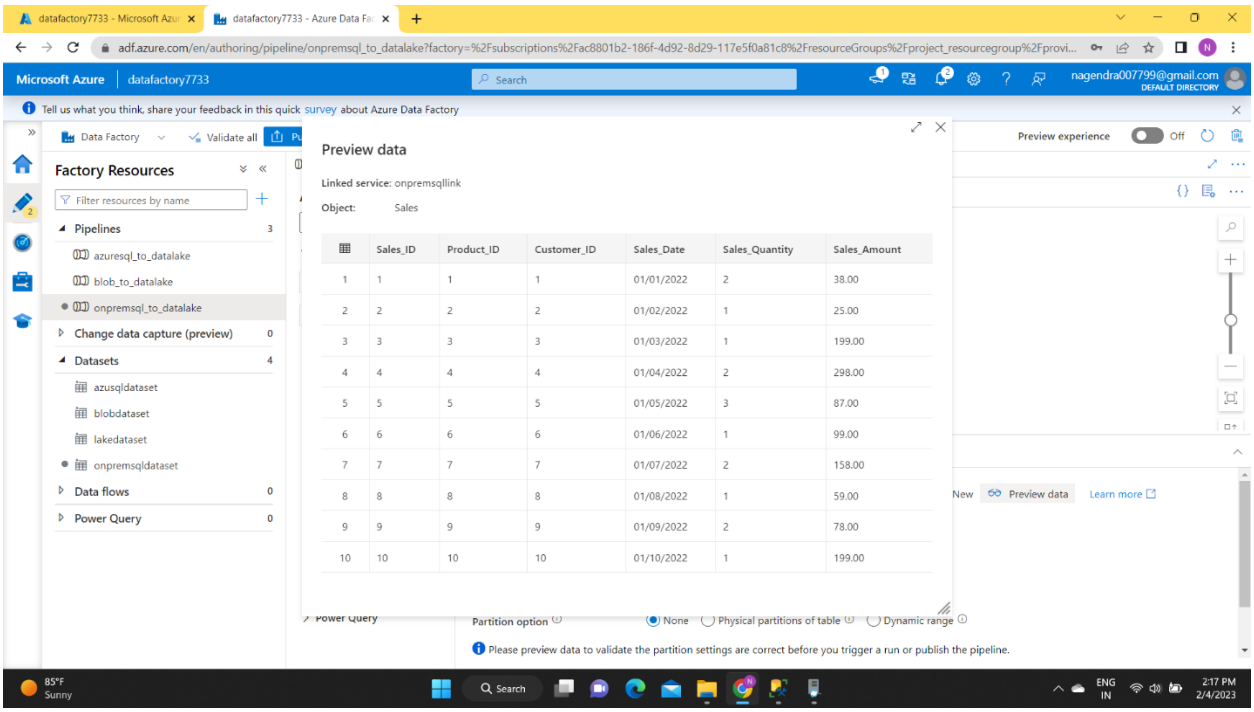
- In dataset select table name and put import schema is none.



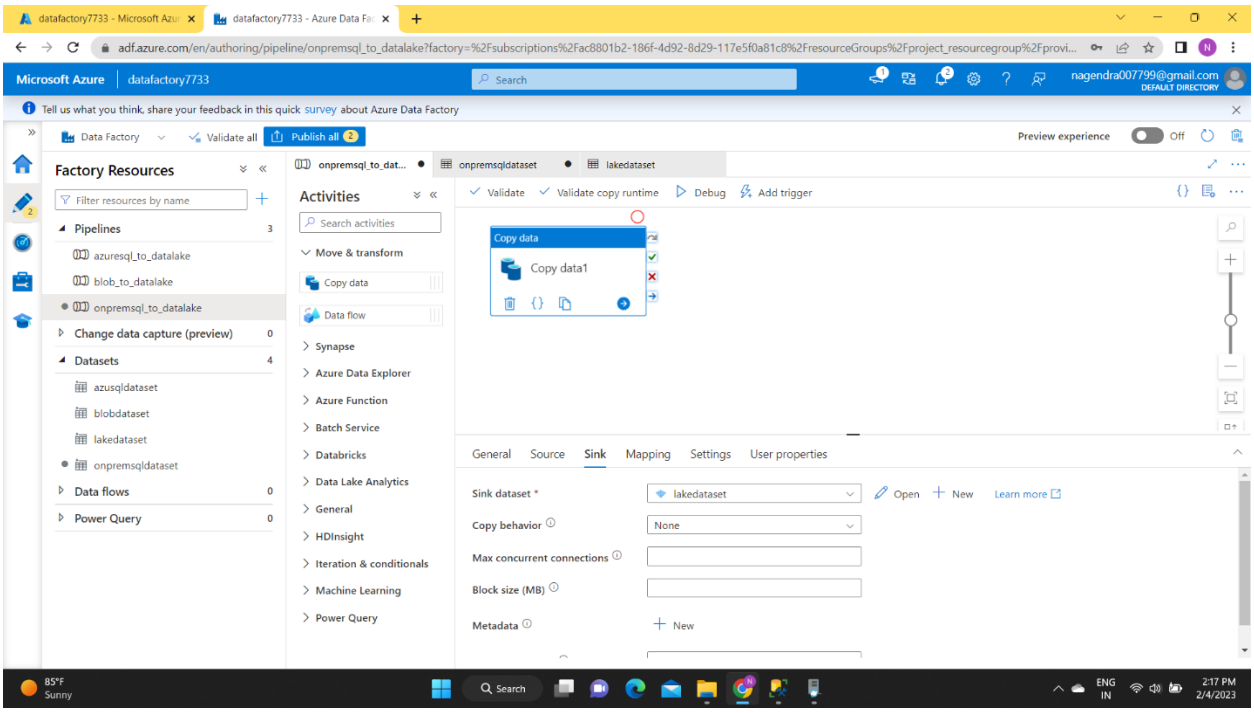
- Remove dbo.sales to sales. By clicking edit button.



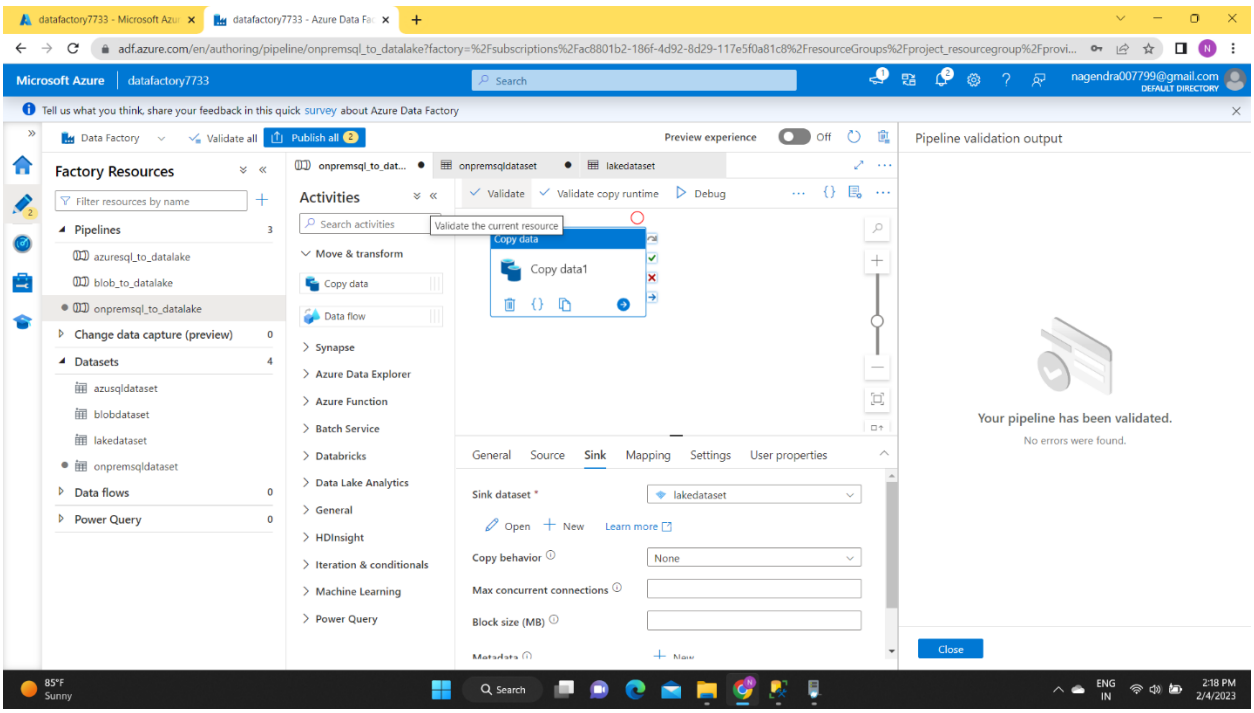
- This is the preview of our sales table.



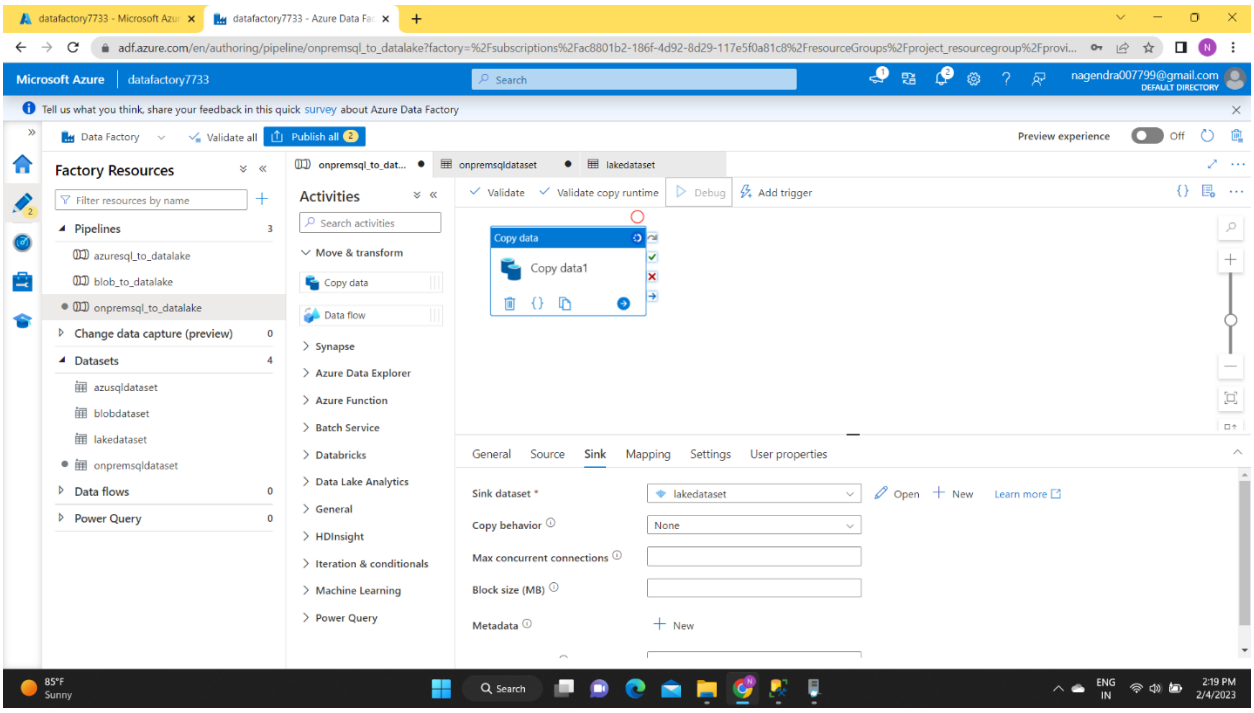
- In sink we have data lake datset .



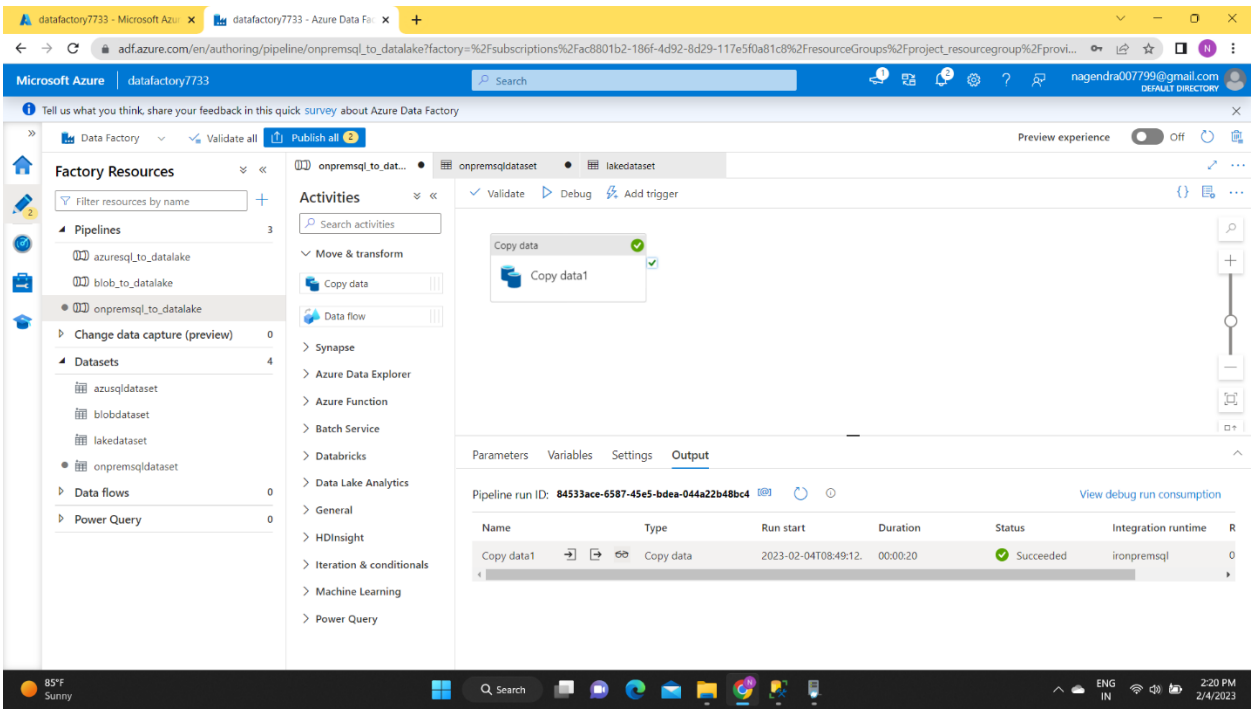
- Validate our pipeline .



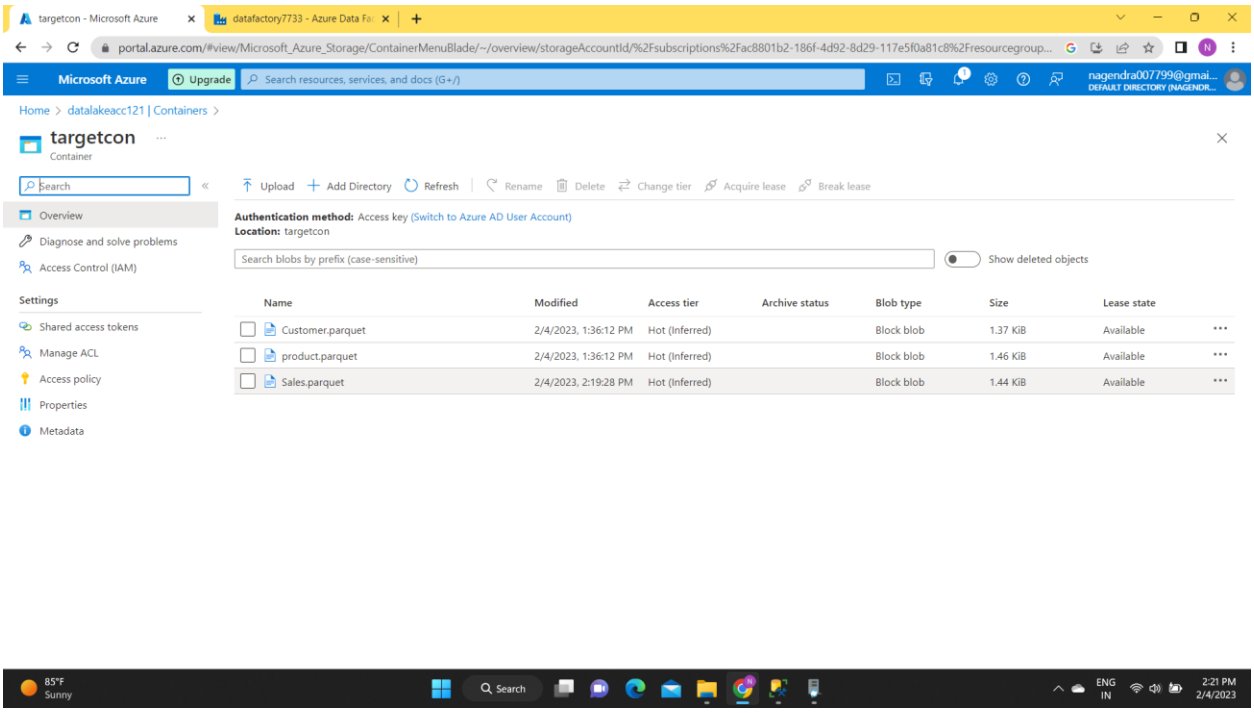
- Now do the debug.



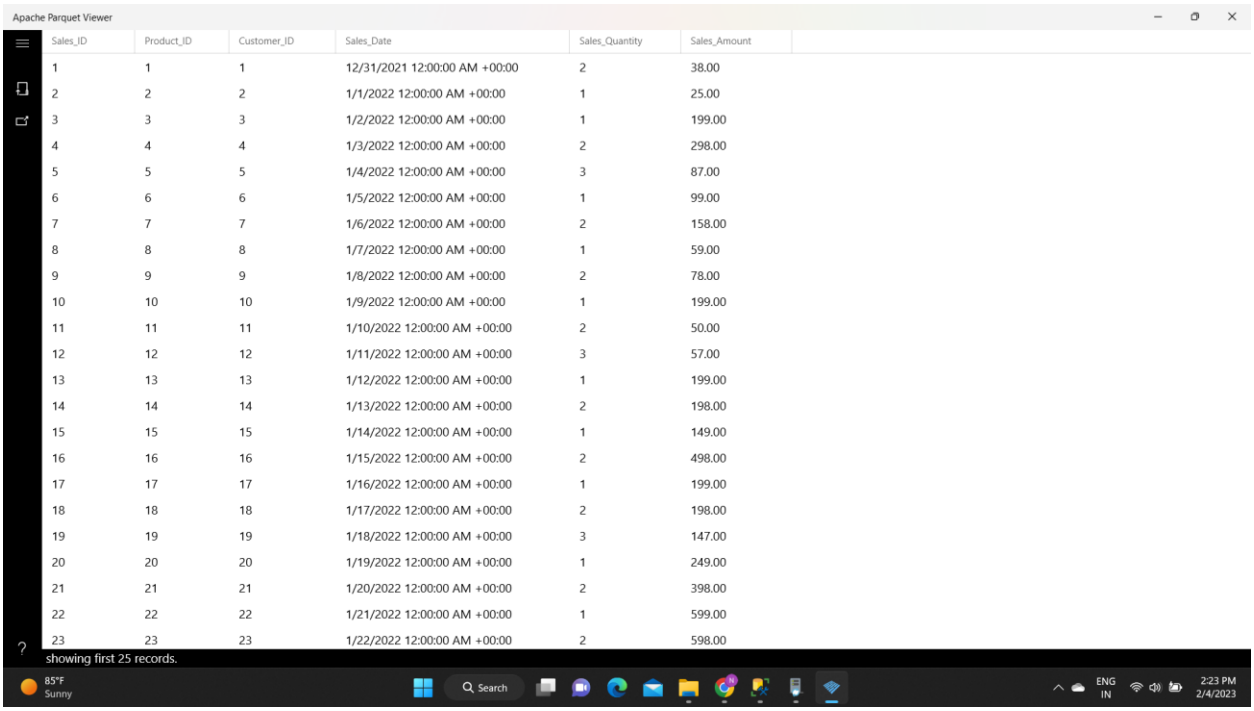
- Our pipeline is succed



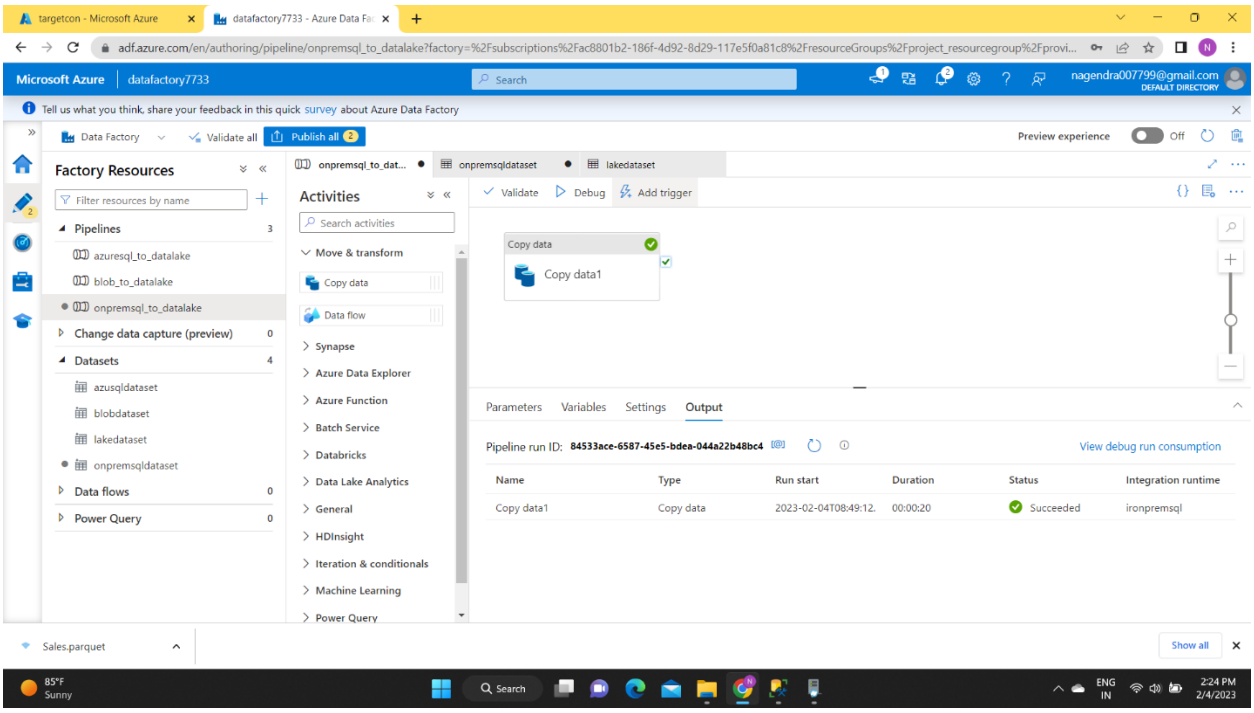
- This is our sales.parquet table in unified data lake.



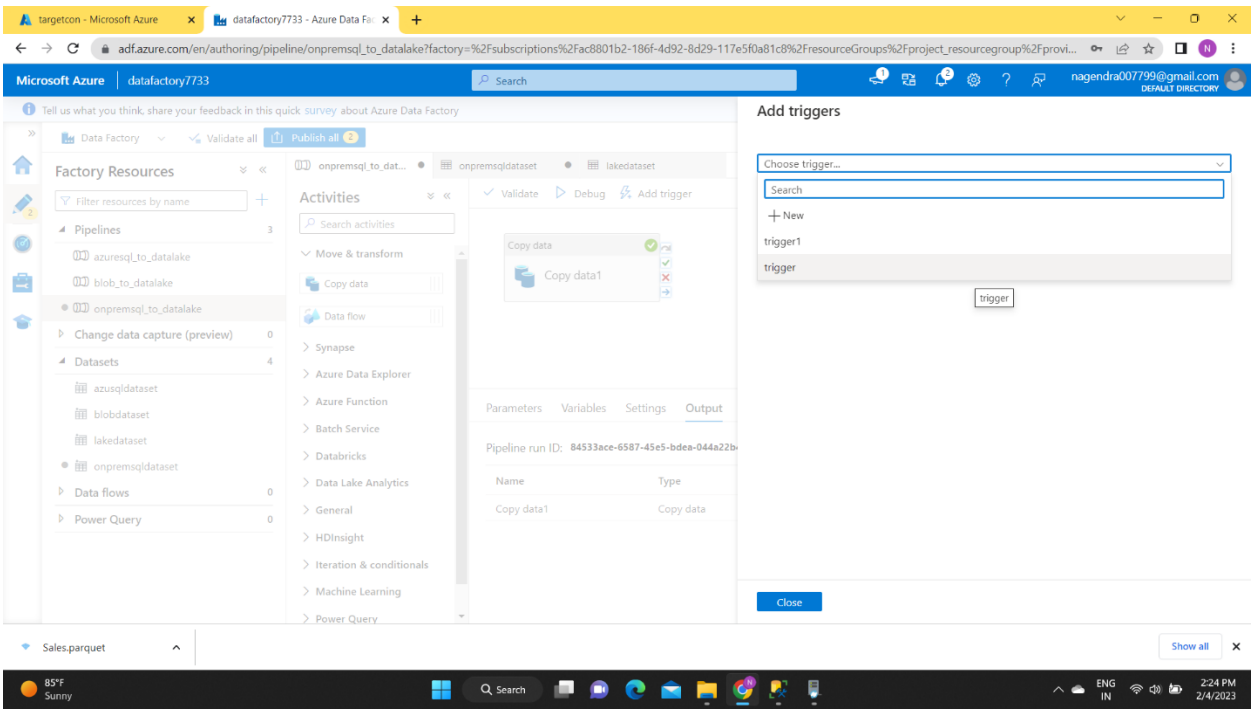
- This is our sales data view.



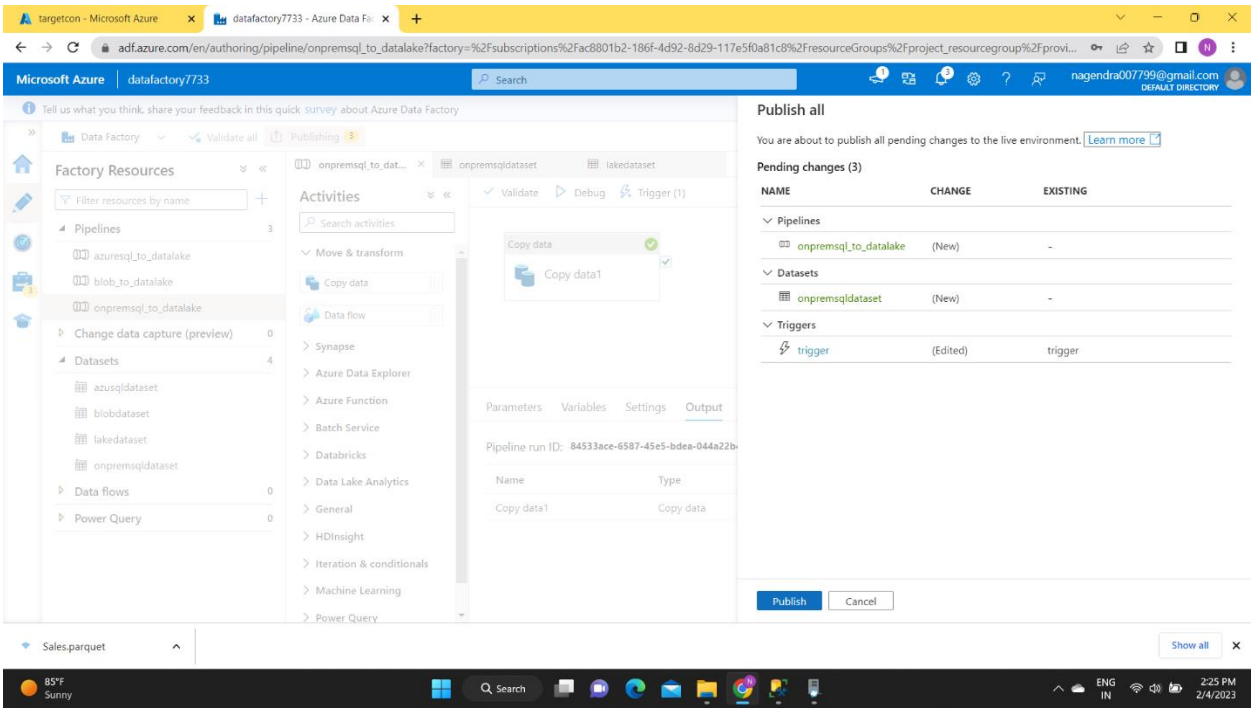
- Now add trigger.



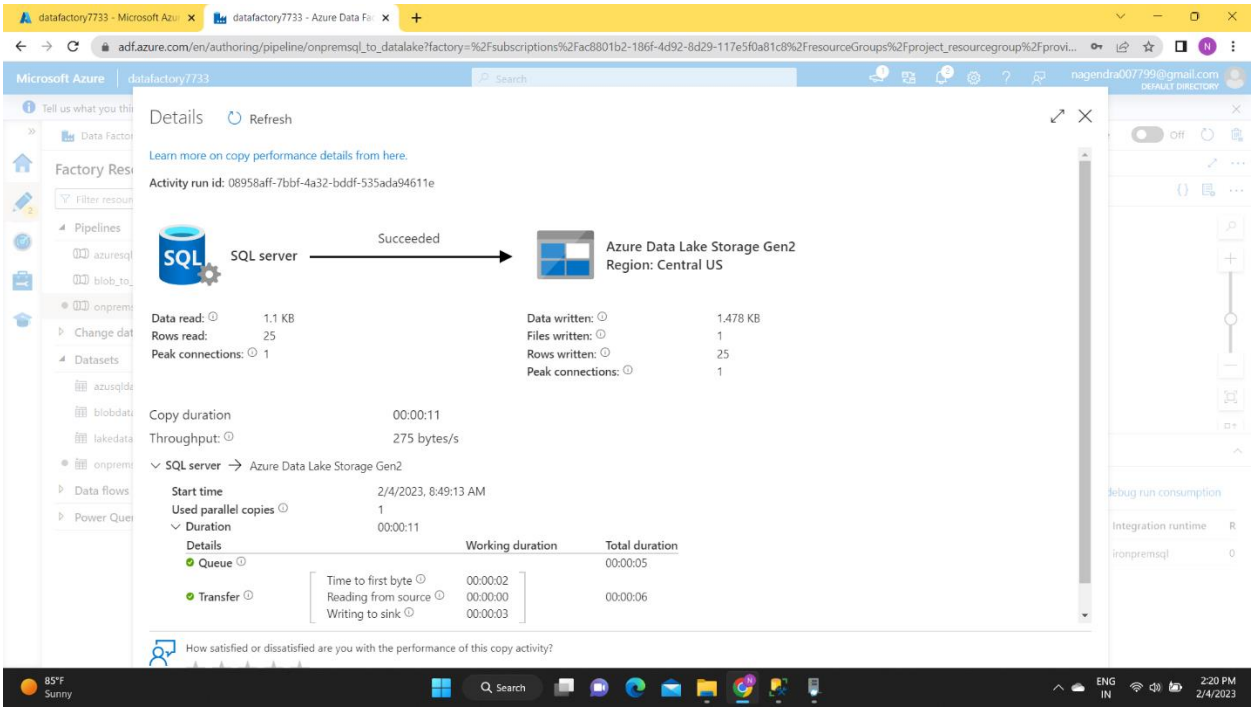
- Here iam giving the previous trigger.



- Now I am going to publish the pipeline.



- This is our on-premises sql server_ to azure data lake pipeline.



- THIS IS OUR UNIFIED STORAGE SYSTEM AND ALL 3 TABLES