1. Pulling data from API
2. Powerful orchestration tool : Azure data factory
3. Building dynamic parameter
4. Pulling data from source and land data in bronze layer
5. bronze(raw) layer: keep data as it is as a replica. Do not apply any transformation.
6. Push data to silver layer with some transformations
7. Serving layer: serving data to stake holders
8. We r making azure warehouse which is Azure synapse analytics
9. Analysis on power bi

Medallion architecture

Medallion Architecture is a layered approach to organizing data processing pipelines, where data flows through multiple stages (or “medallions”) — typically Bronze, Silver, and Gold layers. Each layer refines the data progressively, improving quality and usability.

Layers in Medallion Architecture

1. Bronze Layer (Raw Data)

- Contains raw, unprocessed, or minimally processed data.
- Data is ingested as-is from various sources (logs, IoT devices, databases, streaming sources).
- Usually stores data in its original format.
- May contain duplicates, errors, and inconsistencies.
- Acts as the single source of truth for original data.

2. Silver Layer (Cleaned and Enriched Data)

- Contains cleaned, filtered, and possibly joined data.
- Data quality issues are fixed here (null values handled, invalid records removed).
- Data is structured and normalized for easier querying.
- Can join multiple sources, enrich data with reference datasets.
- Serves as the basis for detailed analytics and business intelligence.

3. Gold Layer (Aggregated and Business-Level Data)

- Contains aggregated, highly curated, and business-friendly datasets.
- Optimized for end-user consumption — dashboards, reports, machine learning models.
- Data here is often denormalized and aggregated by key business dimensions.

- Typically used for KPI reporting, machine learning feature sets, or exporting to other systems.

Why use Medallion Architecture?

- Data Quality Improvement: Step-by-step cleansing and validation.
- Separation of Concerns: Different teams can work independently on each layer.
- Reusability: Silver data can feed multiple gold datasets.
- Scalability: Efficient to process and store data incrementally.
- Traceability: Ability to trace back to raw source data (Bronze) when needed.

Create the resource group
Add tags for doing categorization

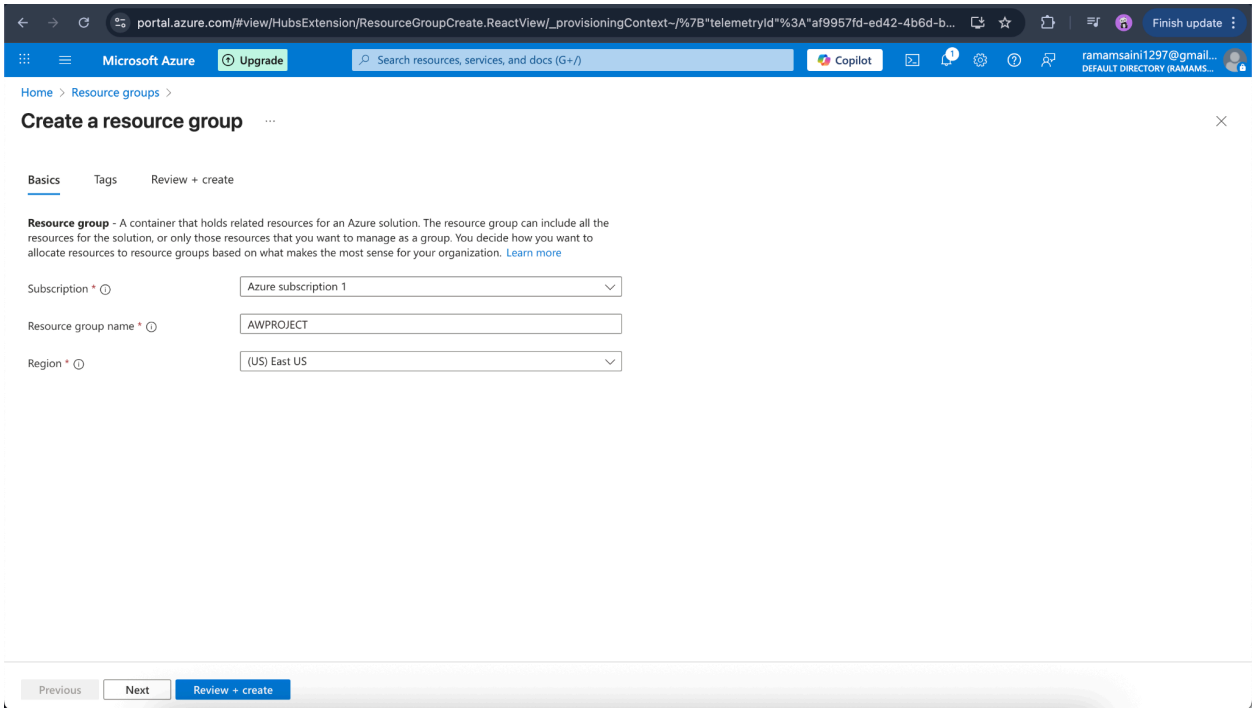
LET'S BEGIN

1. Azure Resource Setup

1.1 Create a Resource Group

A resource group is a container for related Azure resources.

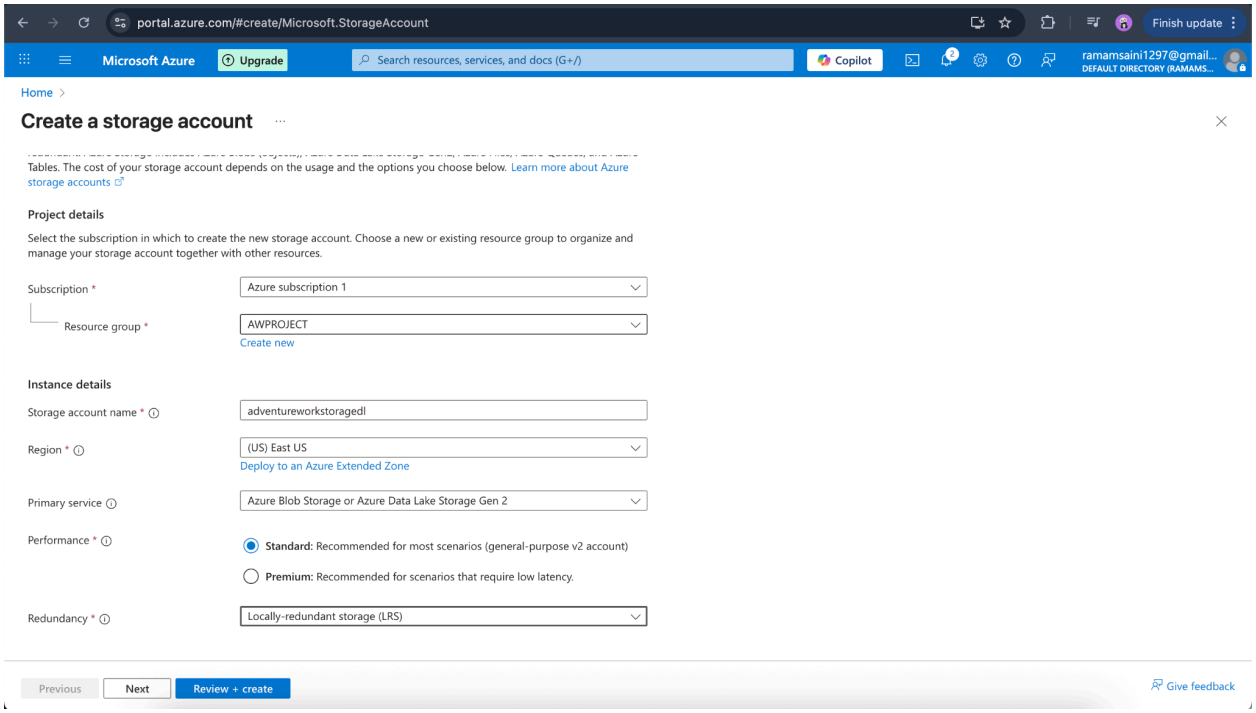
1. Navigate to the Azure portal and create a new Resource Group.
2. Provide a
   Subscription, a Resource group name (e.g., `AWPROJECT`), and a Region.
3. Optionally, add
   tags for categorization.

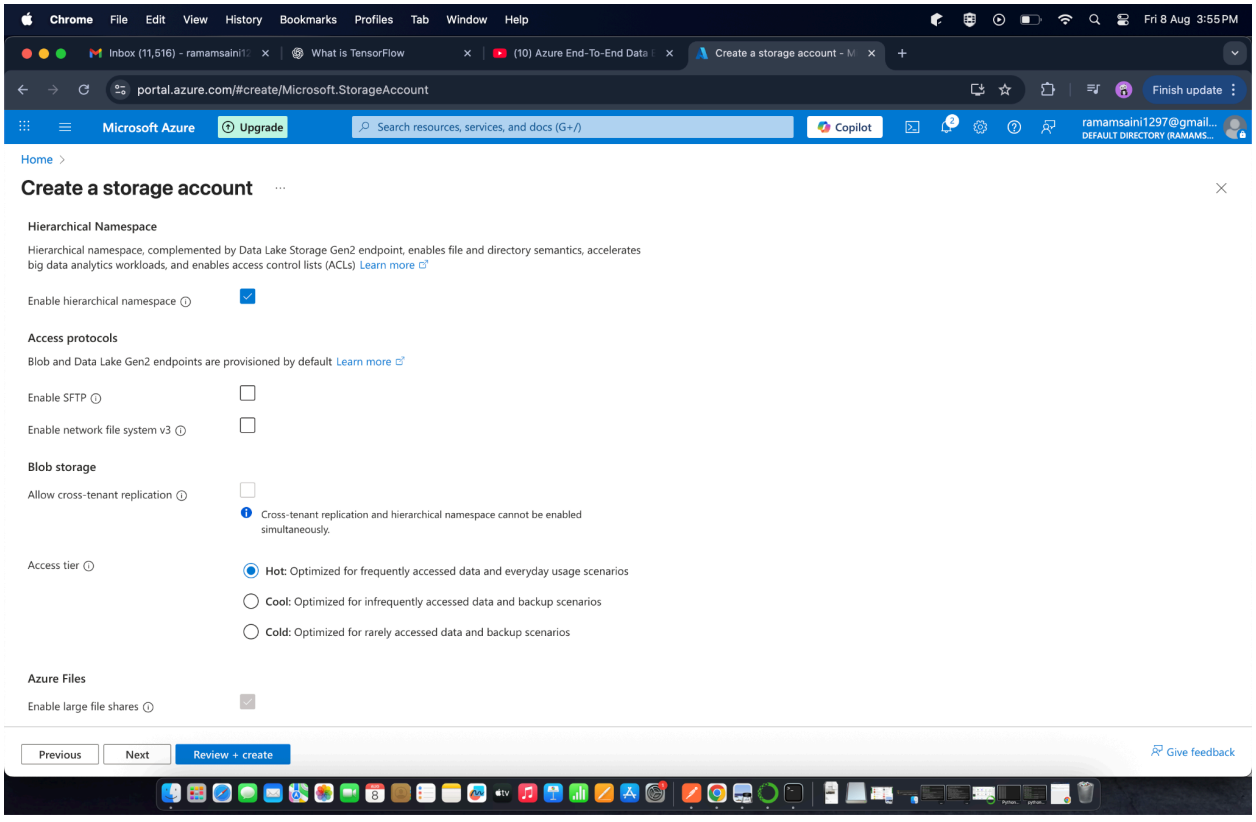


1.2 Create an Azure Data Lake Storage Account

A data lake is a storage account designed for big data analytics. To create one:

1. Create a
   Storage Account.
2. Provide a Subscription, the Resource group created earlier, and a Storage account name (e.g., `adventureworkstoragedl`).
3. Select the Region and Performance tier. For
   Redundancy, Locally-redundant storage (LRS) stores replicas in the same data center, while Geo-redundant storage (GRS) replicates data to different regions.
4. In the
   Advanced tab, enable the Hierarchical namespace to create an Azure Data Lake Gen2 account instead of a standard blob storage account. This allows for a hierarchical file structure with folders.
5. After creation, create containers inside this storage account to represent the layers of the Medallion Architecture:
   `bronze`, `silver`, and `gold`


portal.azure.com/#create/Microsoft.StorageAccount
Finish update
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Copilot
ramamsaini1297@gmail...
DEFAULT DIRECTORY (RAMAMS...
Home
Create a storage account
Tables. The cost of your storage account depends on the usage and the options you choose below. Learn more about Azure storage accounts
Project details
Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.
Subscription *
Azure subscription 1
Resource group *
AWPROJECT
Create new
Instance details
Storage account name *
adventureworkstoragedl
Region *
(US) East US
Deploy to an Azure Extended Zone
Primary service
Azure Blob Storage or Azure Data Lake Storage Gen 2
Performance *
Standard: Recommended for most scenarios (general-purpose v2 account)
Premium: Recommended for scenarios that require low latency.
Redundancy *
Locally-redundant storage (LRS)
Previous
Next
Review + create
Give feedback


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Fri 8 Aug 3:55 PM
Inbox (11,516) - ramamsaini1
What is TensorFlow
(10) Azure End-To-End Data
Create a storage account - M
portal.azure.com/#create/Microsoft.StorageAccount
Finish update
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Copilot
ramamsaini1297@gmail...
DEFAULT DIRECTORY (RAMAMS...
Home
Create a storage account
Hierarchical Namespace
Hierarchical namespace, complemented by Data Lake Storage Gen2 endpoint, enables file and directory semantics, accelerates big data analytics workloads, and enables access control lists (ACLs) Learn more
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
Allow cross-tenant replication
Cross-tenant replication and hierarchical namespace cannot be enabled simultaneously.
Access tier
Hot: Optimized for frequently accessed data and everyday usage scenarios
Cool: Optimized for infrequently accessed data and backup scenarios
Cold: Optimized for rarely accessed data and backup scenarios
Azure Files
Enable large file shares
Previous
Next
Review + create
Give feedback

Use data lake when need a hierarchical file structure (folders).

Azure Data Factory

Azure Data Factory is a cloud-based data integration service provided by Microsoft Azure. It allows you to create, schedule, and orchestrate data pipelines to move and transform data from various sources to your desired destination — typically for analytics or storage.

Azure Data Factory = ETL / ELT tool on Azure

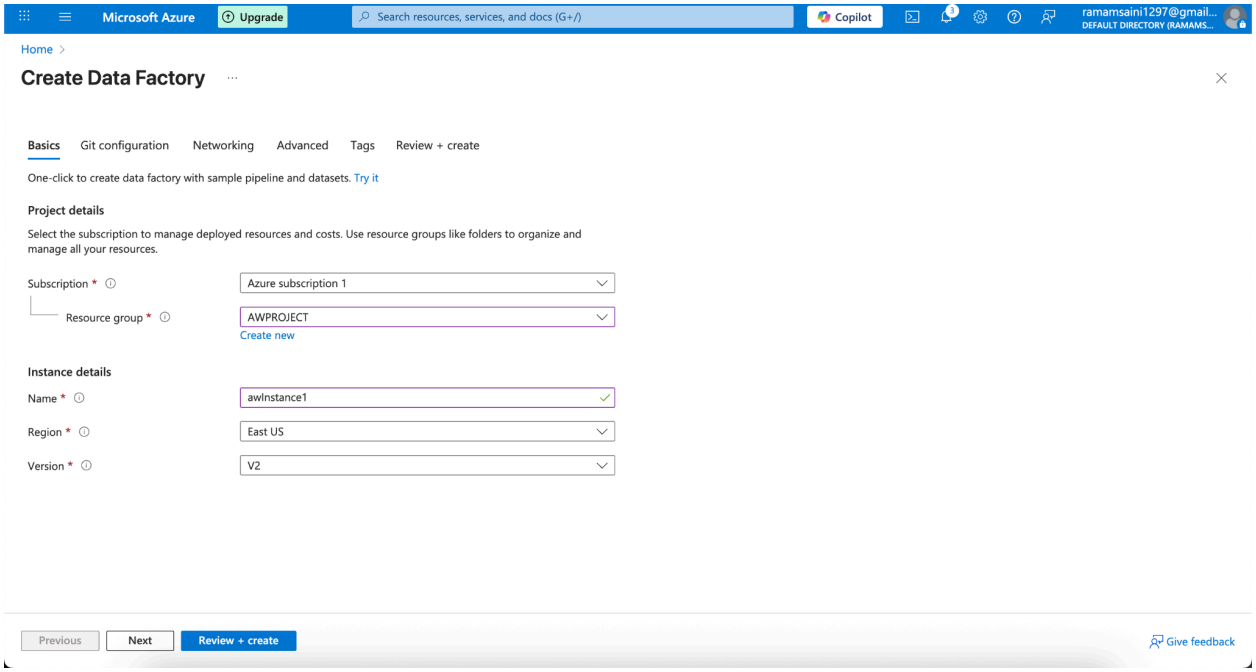
- Extract → from data sources (e.g., SQL, APIs, blob storage, on-prem)
- Transform → using data flows or external compute like Azure Databricks
- Load → into destinations like Azure Synapse, Data Lake, SQL Database, etc.

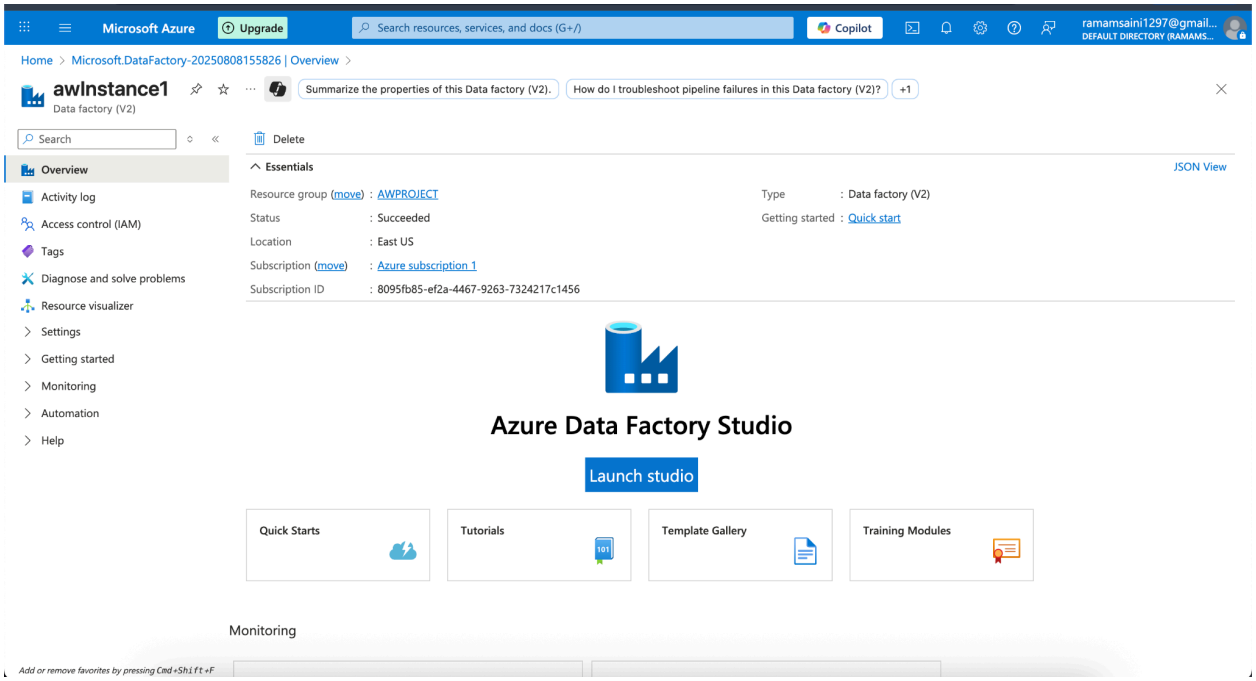
Crete containers that indicates for zones (bronze, silver and gold)
Crete the zones inside azure data factory (adventureworkstorage)

2.2 Creating a Static Pipeline

A static pipeline is a simple, non-dynamic data copy process.

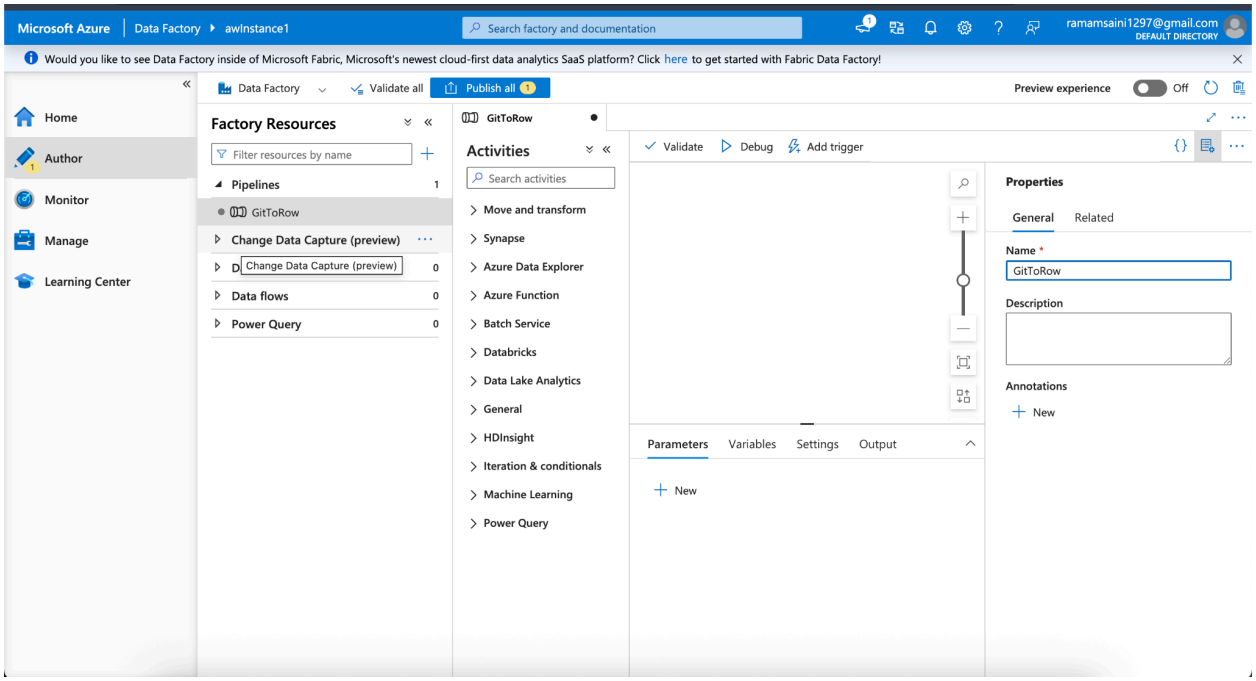
1. Create an Azure Data Factory instance in the `AWPROJECT` resource group.
2. In the ADF Studio, navigate to the Manage section to create Linked Services.
3. Create an HTTP linked service to connect to the source API (e.g., GitHub).
4. Create a second linked service to connect to the Azure Data Lake Storage Gen2 account (`storageDataLake`).
5. In the Author section, create a new pipeline.
6. Add a Copy data activity to the pipeline.
7. Configure the Source dataset, using the HTTP linked service and a relative URL to the CSV file.
8. Configure the Sink dataset, using the data lake linked service. Set the File path to the `bronze` container with a subfolder (e.g., `/bronze/products/products.csv`).
9. Debug the pipeline to ensure the data is successfully copied to the data lake.
10. Publish the changes to save your work. The data should now be in the `bronze` layer of your data lake. This is the raw data, kept as a replica without any transformations.





Author : creating all pipelines
Monitor: monitor our pipelines, failed pipelines , reasons of failed pipelines
Manage:manage things like repo github connections, devops connections and link services
Learning center: read resources, documentation pick avail. Data sets

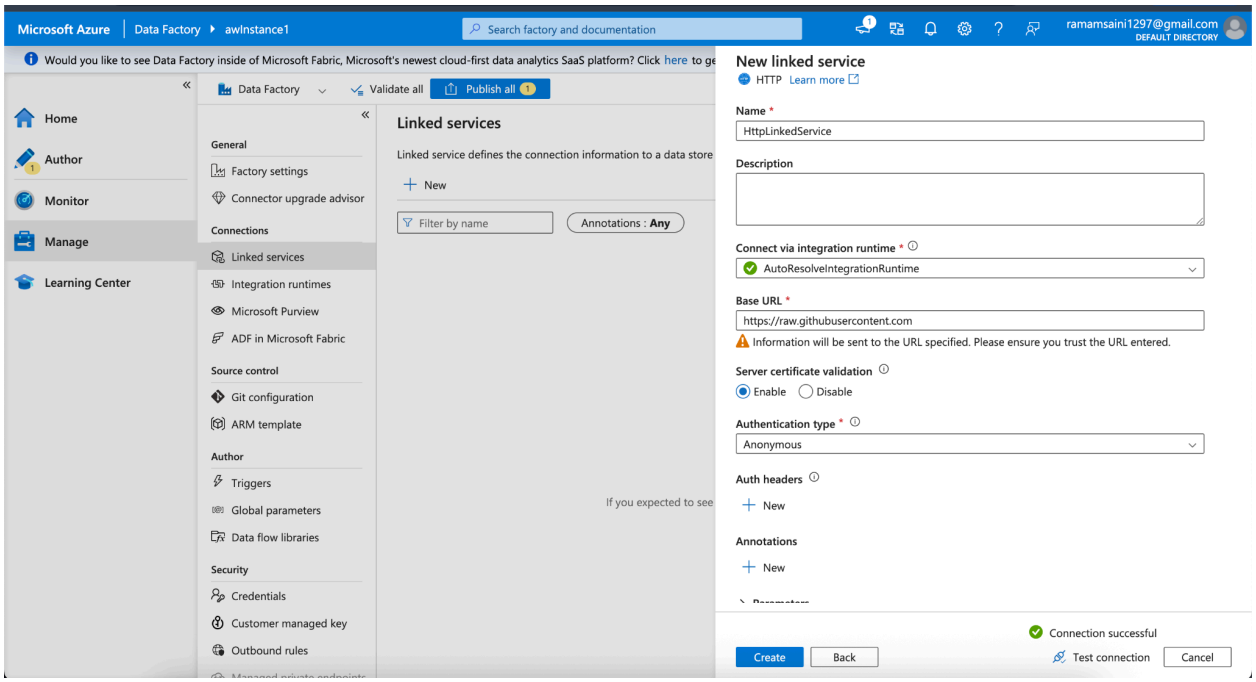
Craete the containers for all zones in data lake (adentureworkstoragedl)

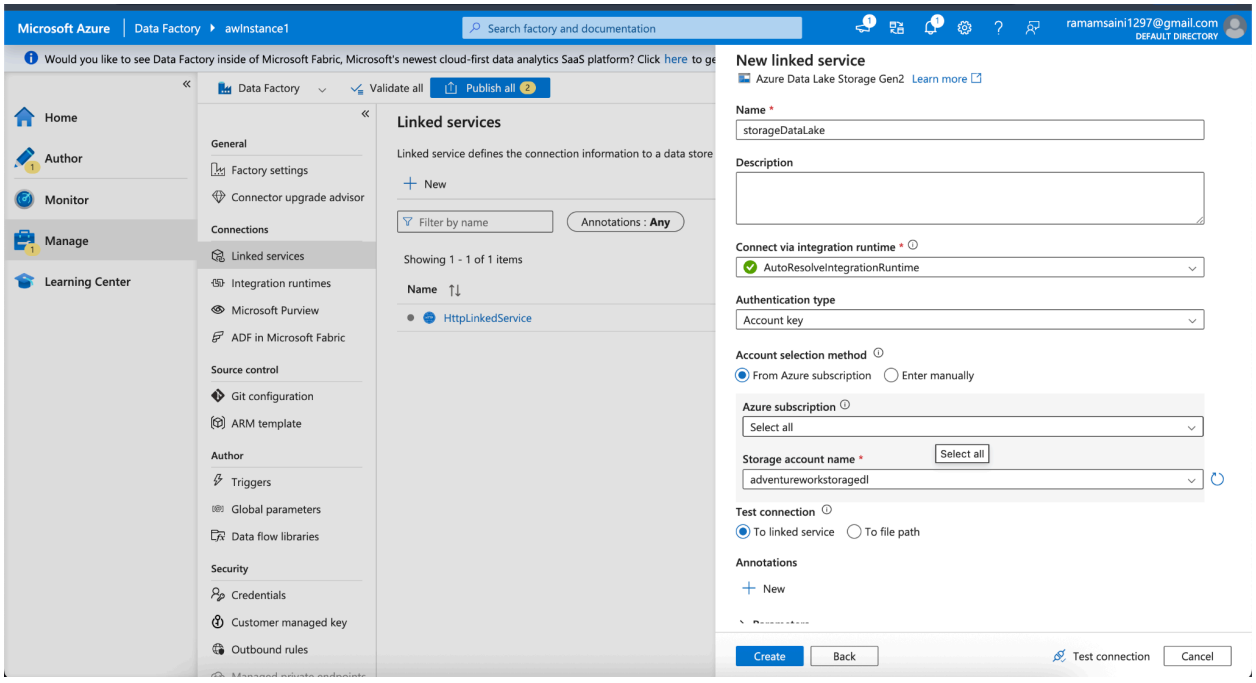


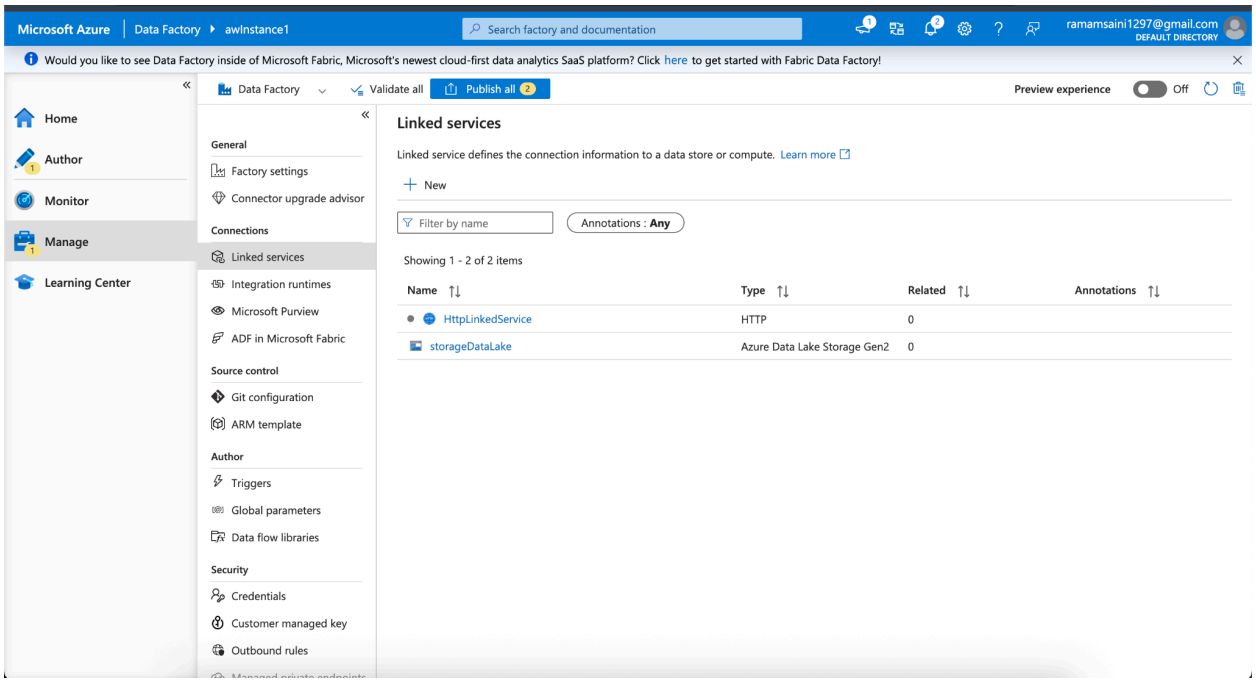
ETL pipeline with Azure Data Factory

Link services: connection that raw data has build with source and destination

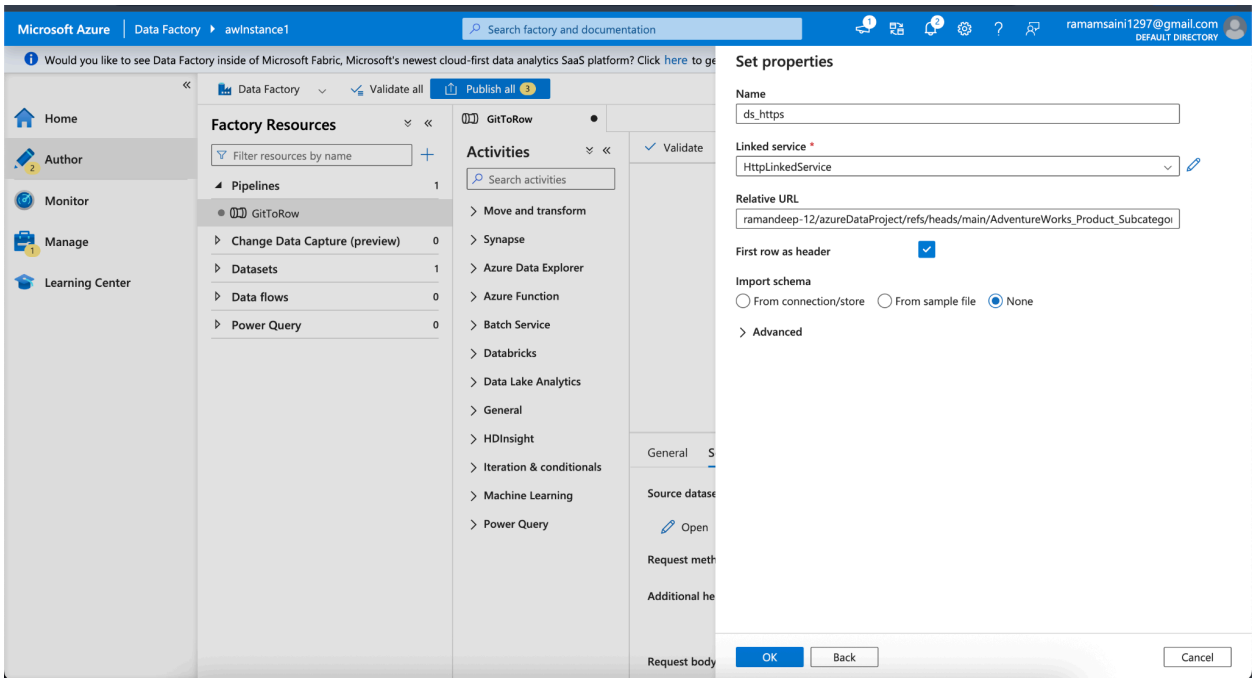
Put Data from giithub and add in data lake
Create 2 linked services
Like pick data from http and push to data lake
Http linked service as taking data from git hub

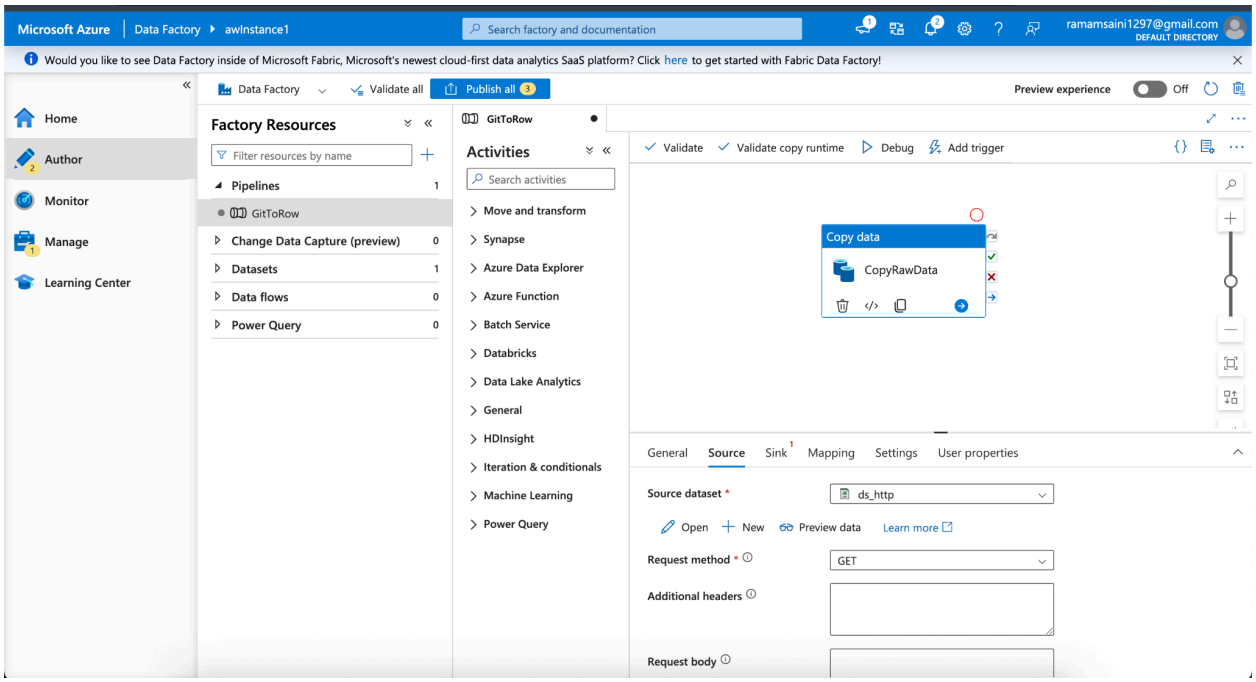

Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click here to ge
Data Factory
Validate all
Publish all 1
Home
Author
Monitor
Manage
Learning Center
General
Factory settings
Connector upgrade advisor
Connections
Linked services
Integration runtimes
Microsoft Purview
ADF in Microsoft Fabric
Source control
Git configuration
ARM template
Author
Triggers
Global parameters
Data flow libraries
Security
Credentials
Customer managed key
Outbound rules
Linked services
Linked service defines the connection information to a data store
New
Filter by name
Annotations : Any
If you expected to see
New linked service
HTTP Learn more
Name *
HttpLinkedService
Description
Connect via integration runtime *
AutoResolveIntegrationRuntime
Base URL *
https://raw.githubusercontent.com
Information will be sent to the URL specified. Please ensure you trust the URL entered.
Server certificate validation
Enable
Disable
Authentication type *
Anonymous
Auth headers
New
Annotations
New
Connection successful
Create
Back
Test connection
Cancel


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click here to ge
Data Factory
Validate all
Publish all 2
Home
Author
Monitor
Manage
Learning Center
General
Factory settings
Connector upgrade advisor
Connections
Linked services
Integration runtimes
Microsoft Purview
ADF in Microsoft Fabric
Source control
Git configuration
ARM template
Author
Triggers
Global parameters
Data flow libraries
Security
Credentials
Customer managed key
Outbound rules
Linked services
Linked service defines the connection information to a data store
New
Filter by name
Annotations : Any
Showing 1 - 1 of 1 items
Name
HttpLinkedService
New linked service
Azure Data Lake Storage Gen2 Learn more
Name *
storageDataLake
Description
Connect via integration runtime *
AutoResolveIntegrationRuntime
Authentication type
Account key
Account selection method
From Azure subscription
Enter manually
Azure subscription
Select all
Storage account name *
Select all
adventureworkstoragedl
Test connection
To linked service
To file path
Annotations
New
Create
Back
Test connection
Cancel



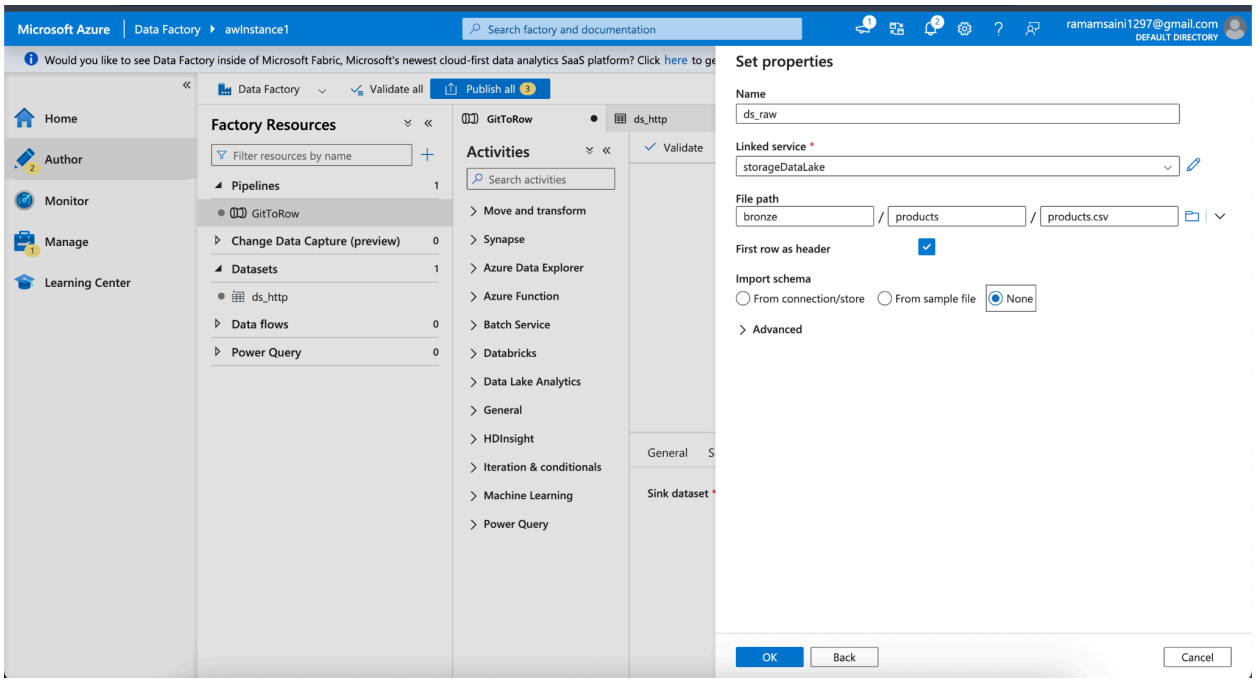
Create the data set out of it
Created the static pipeline
Create the source data set as
author—> source—-> New—> http—> csv



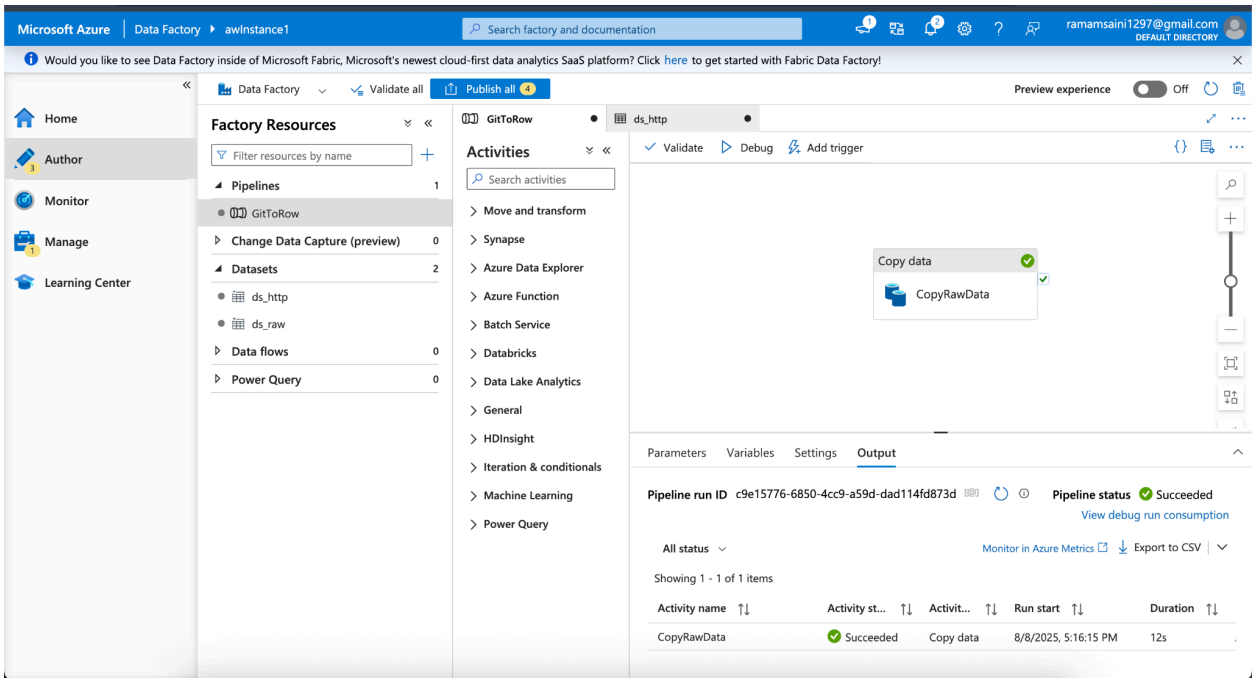


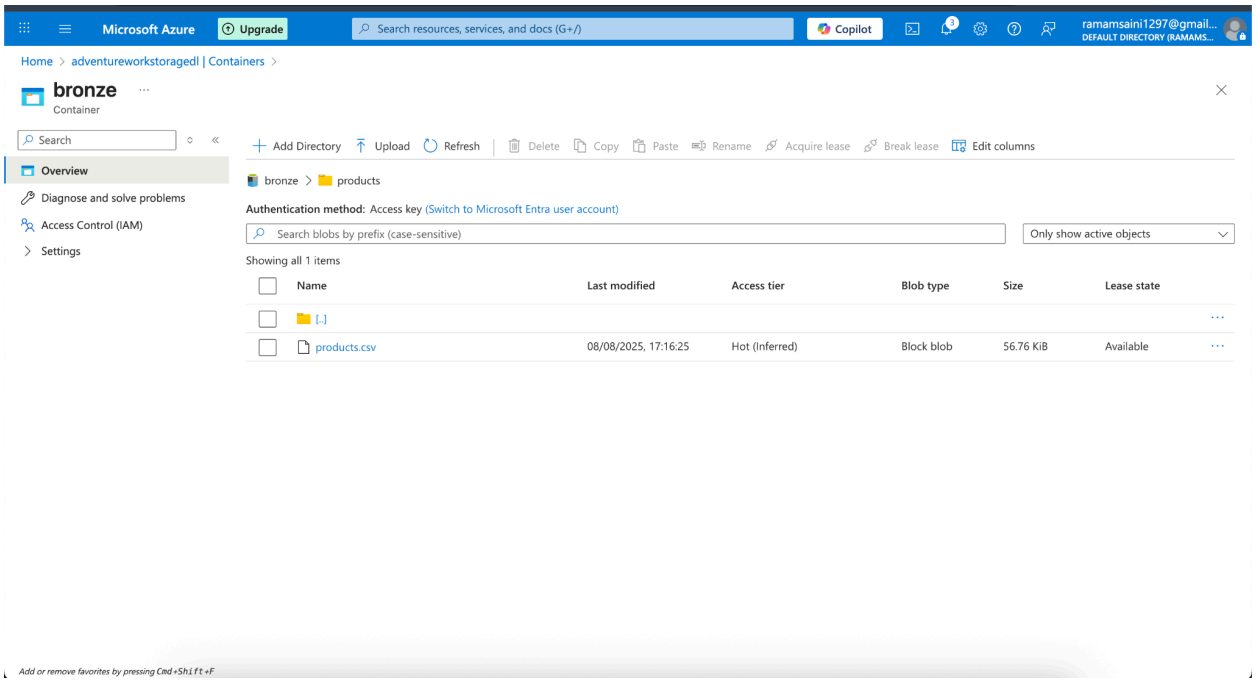
For sink

New→ data lake gen 2—> csv



Debug





2.3 Creating a Dynamic Pipeline

A dynamic pipeline handles multiple files or changing parameters automatically.

1. Create a new container in your data lake, named `parameters`.

2. Upload a JSON file (e.g., `dummy.json`) to this container. This file will contain the relative URL, folder name, and file name for each source file.
3. In the ADF pipeline, add a Lookup activity to read the contents of the `dummy.json` file.
4. Add a For Each activity and set its Items property to the output of the Lookup activity using the expression `@activity('LookupGit').output.value`.
5. Inside the For Each activity, add the Copy data activity.
6. For the Source dataset, create a dynamic dataset with a parameter for the relative URL. Set the value to `@item().parameter_relative_url`.
7. For the Sink dataset, create a dynamic dataset with parameters for the sink folder and file name. Set their values to `@item().p_sink_folder` and `@item().p_sink_fileName` respectively.

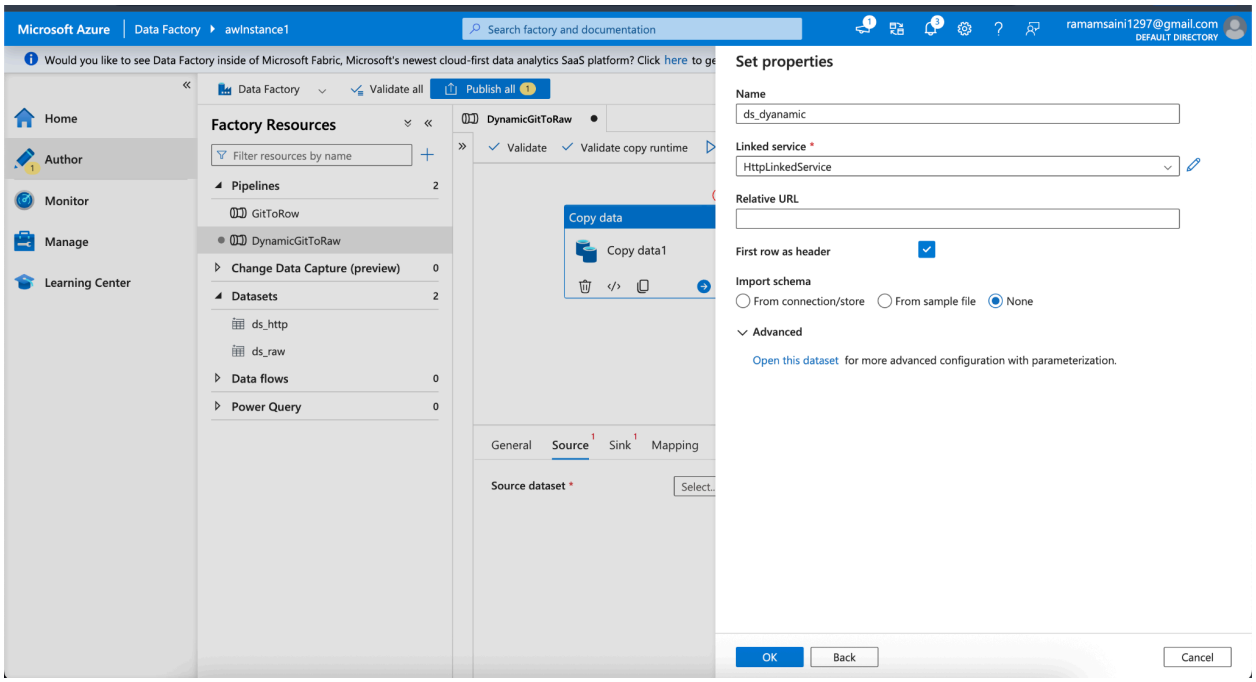
Publish this as it saves our work in data factory and whenever we come then able to see all the progress otherwise we need to rebuild everything

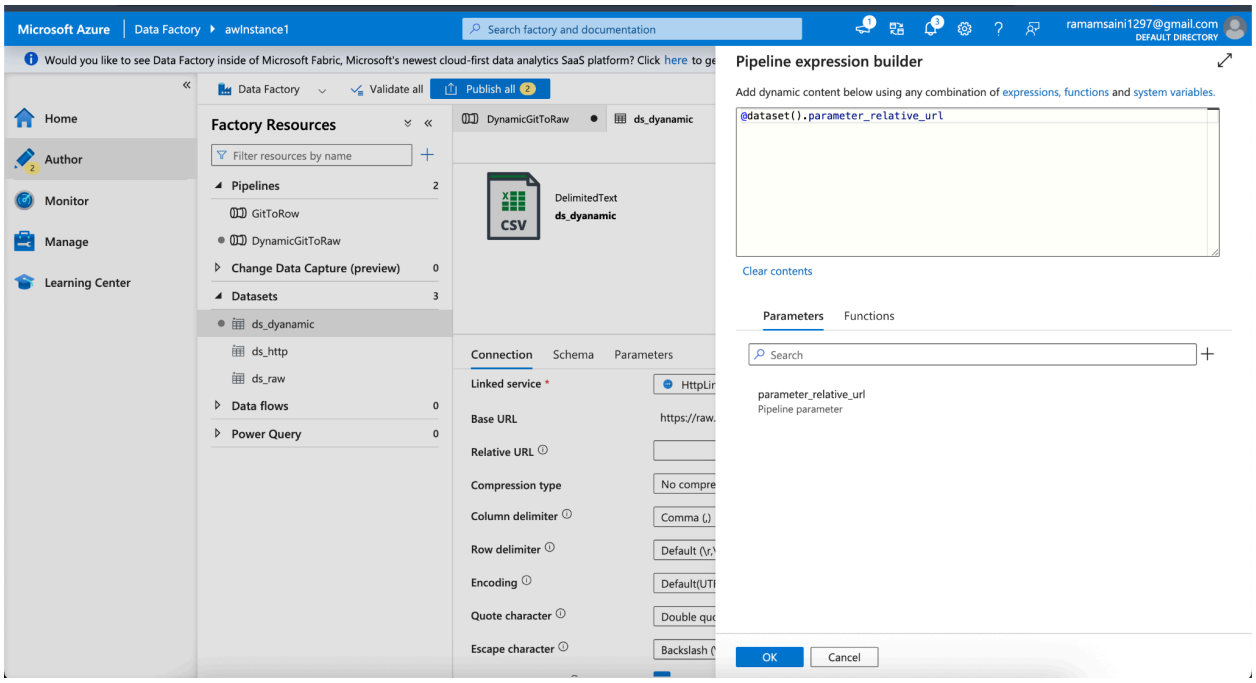
Dynamic Pipeline
Using iterations and conditions

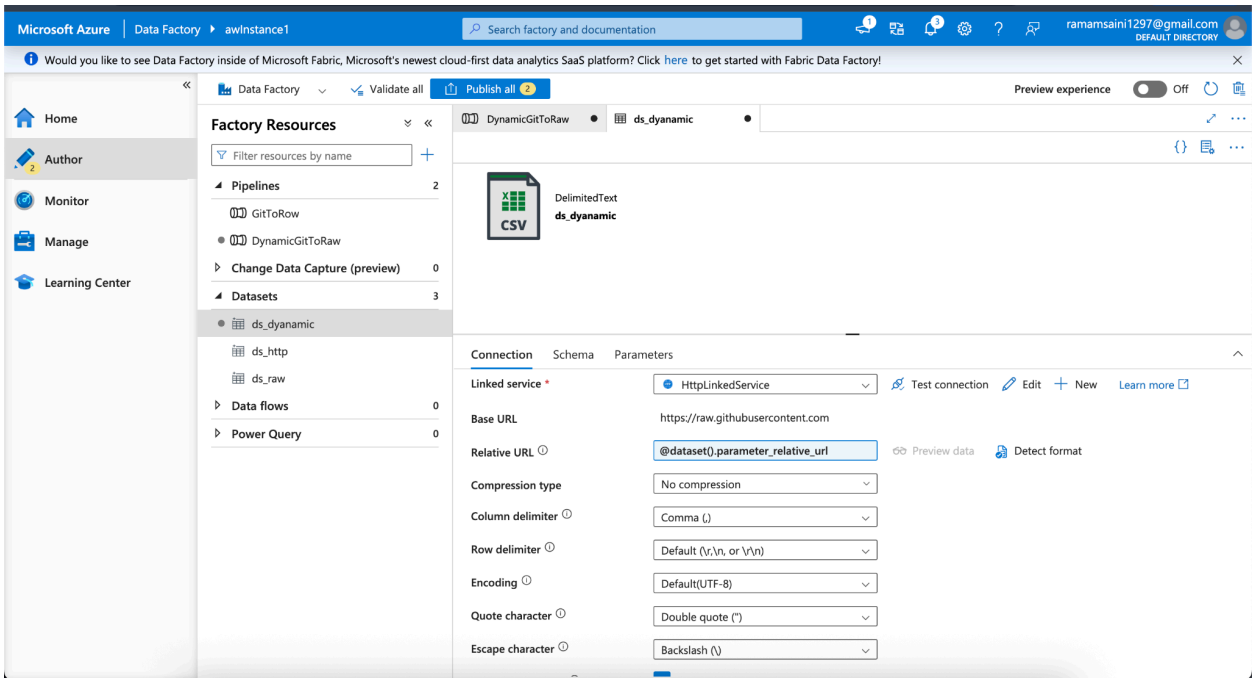
1. Relative url
2. Folder where we store data
3. File name
   Above 3 got changes everytime so instead of changing manually we create the parameters

For each Activity(for loop)
Put copy activity inside for each activity these things keep on moving till our iterations got completed

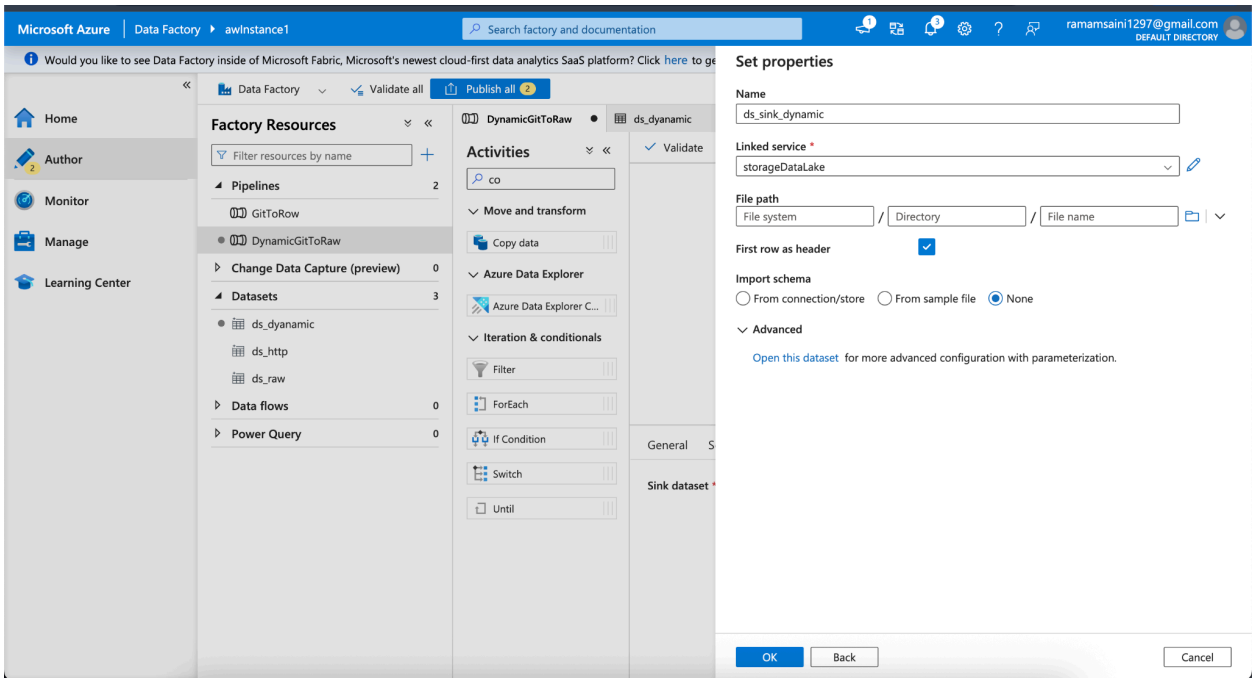
Need one data set that contains data of all the files… one data set to pick all the data set instead of creating multiple data sets called parameterization

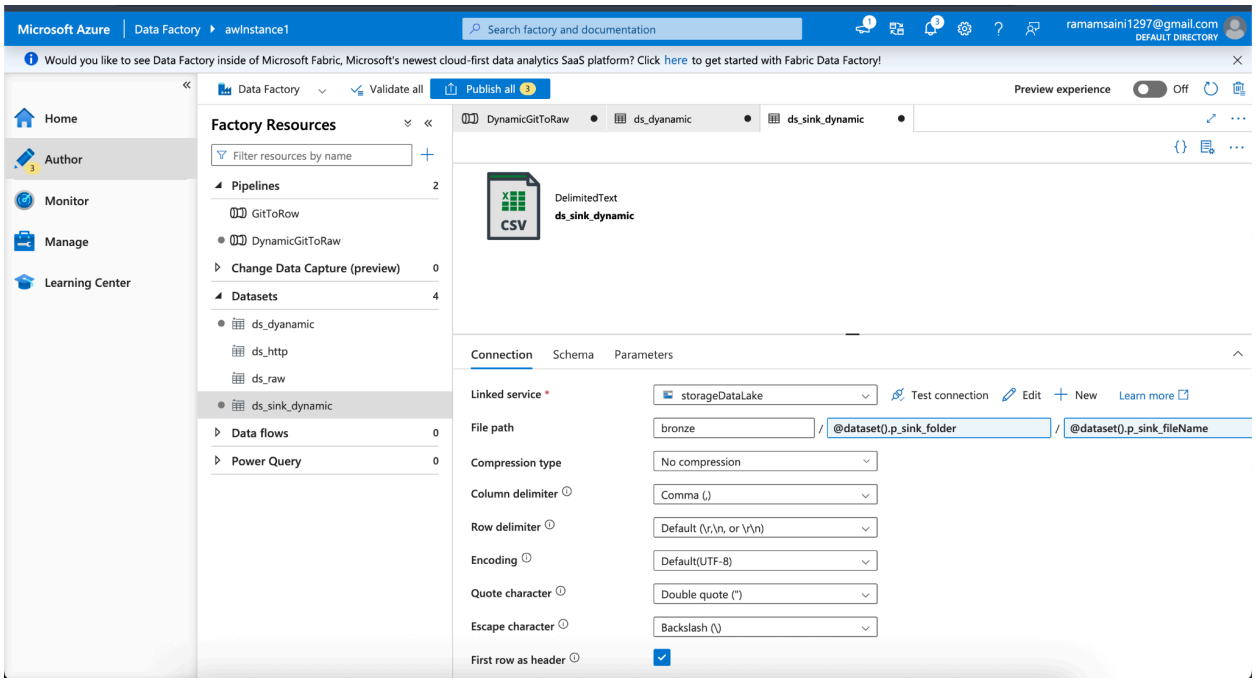

Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click here to ge
Home
Author
Monitor
Manage
Learning Center
Data Factory
Validate all
Publish all 1
Factory Resources
Filter resources by name
Pipelines 2
GitToRow
DynamicGitToRaw
Change Data Capture (preview) 0
Datasets 2
ds_http
ds_raw
Data flows 0
Power Query 0
DynamicGitToRaw
Validate
Validate copy runtime
Copy data
Copy data1
General
Source 1
Sink 1
Mapping
Source dataset *
Select..
Set properties
Name
ds_dyanamic
Linked service *
HttpLinkedService
Relative URL
First row as header
Import schema
From connection/store
From sample file
None
Advanced
Open this dataset for more advanced configuration with parameterization.
OK
Back
Cancel


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click here to ge
Home
Author
Monitor
Manage
Learning Center
Data Factory
Validate all
Publish all 2
Factory Resources
Filter resources by name
Pipelines 2
GitToRow
DynamicGitToRaw
Change Data Capture (preview) 0
Datasets 3
ds_dyanamic
ds_http
ds_raw
Data flows 0
Power Query 0
DynamicGitToRaw
ds_dyanamic
CSV
DelimitedText
ds_dyanamic
Connection
Schema
Parameters
Linked service *
Base URL
Relative URL
Compression type
Column delimiter
Row delimiter
Encoding
Quote character
Escape character
Pipeline expression builder
Add dynamic content below using any combination of expressions, functions and system variables.
@dataset().parameter_relative_url
Clear contents
Parameters
Functions
Search
parameter_relative_url
Pipeline parameter
OK
Cancel



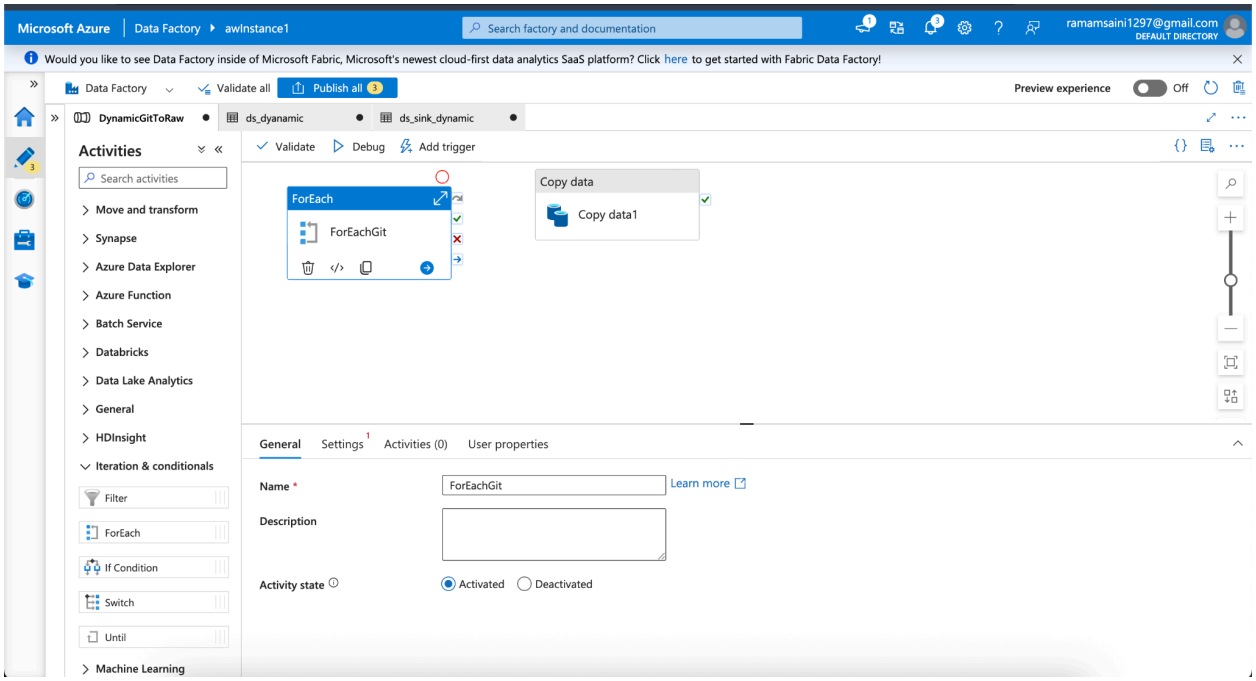
Created sink dynamic parameter

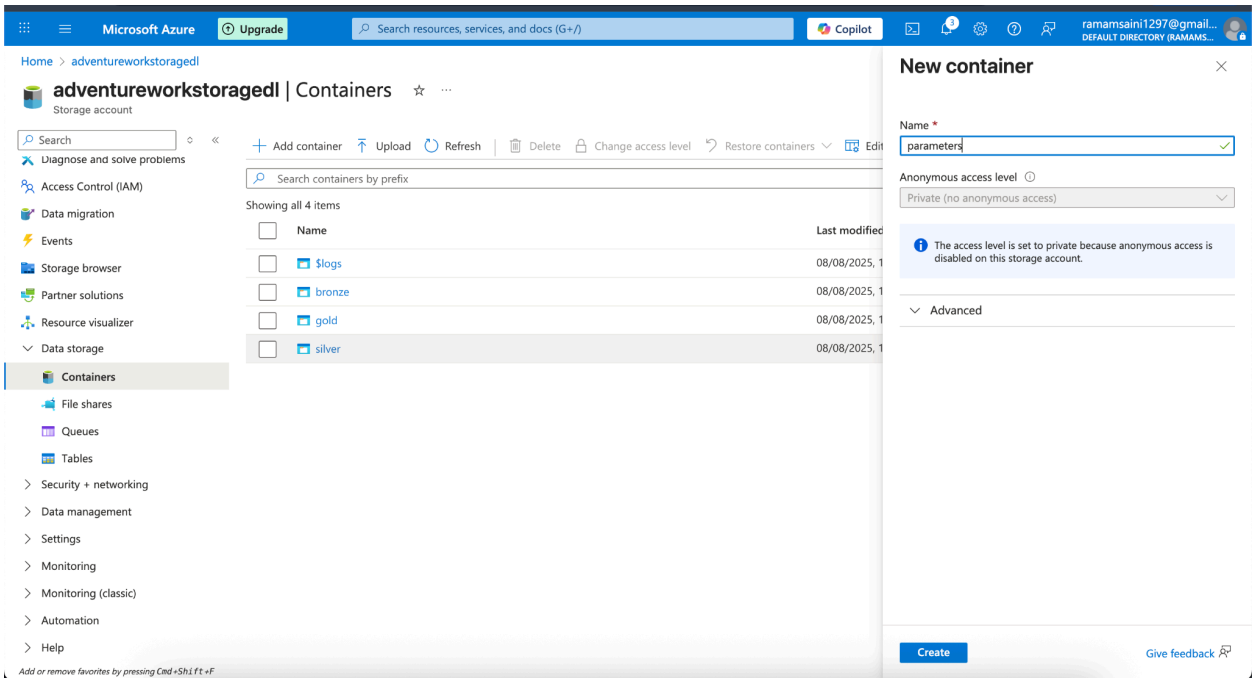


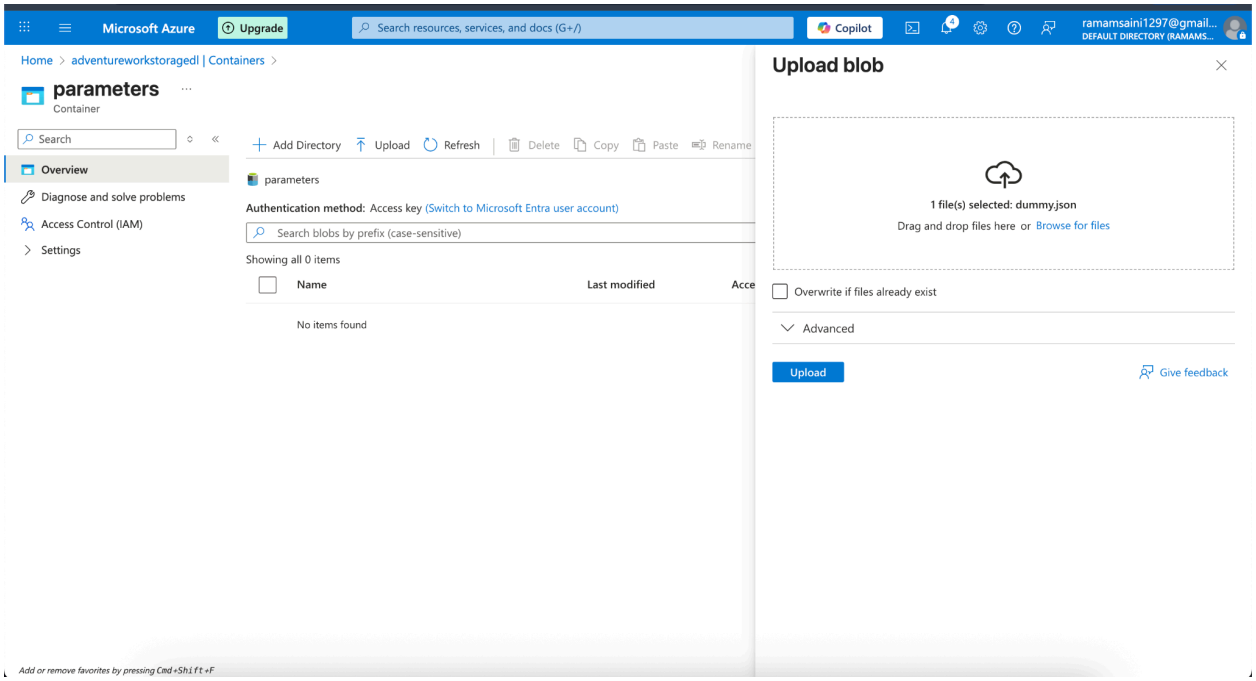


For each activity
In items need the entities which should be run to properly

Make one parameter container where upload one json file which contains all the relative_url , foldername and file name



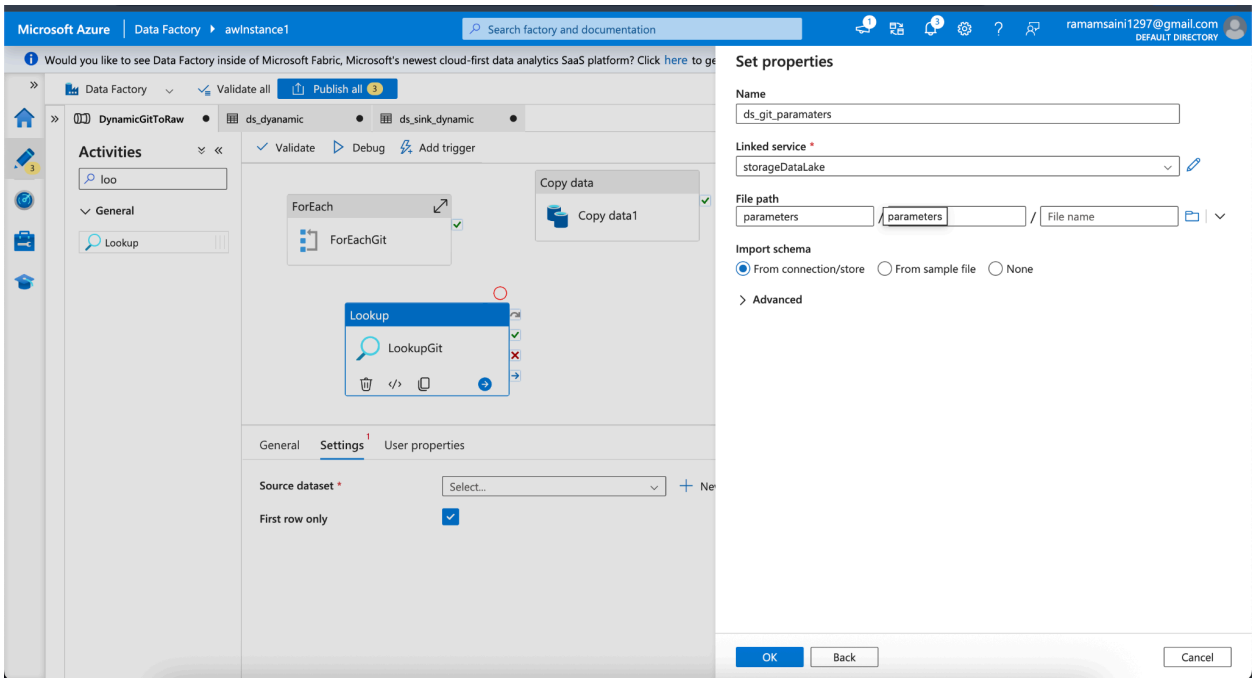




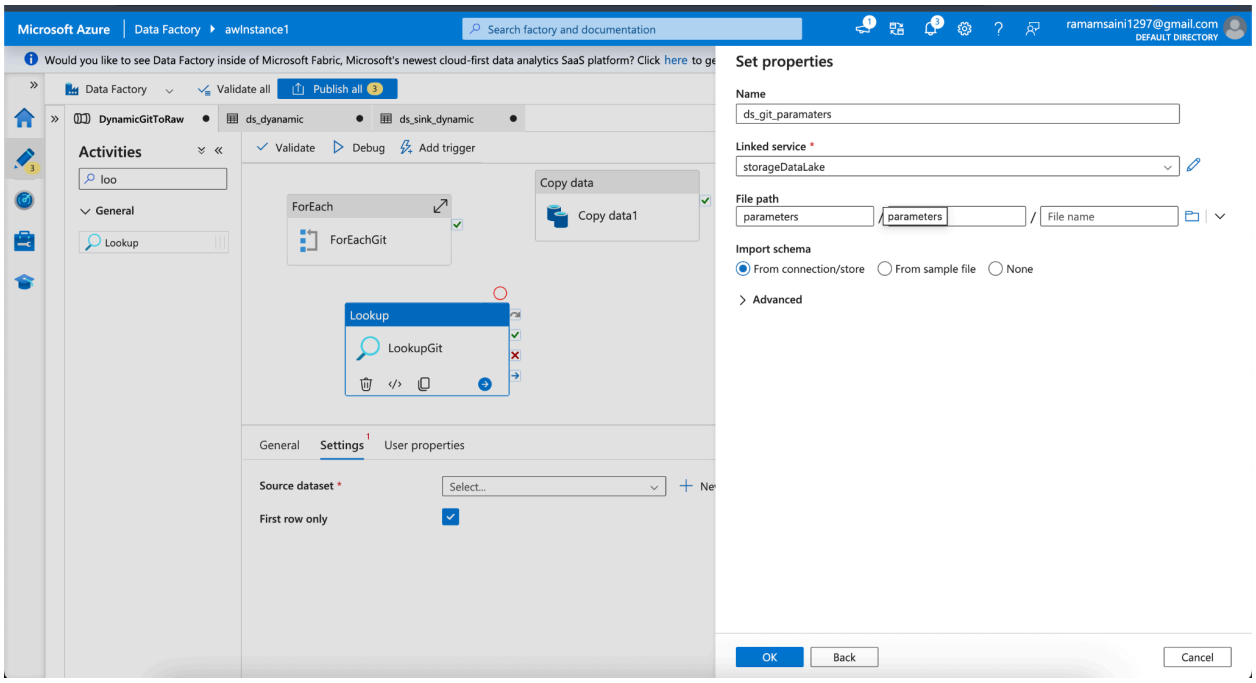
Lookup activity (gives the output of the data)

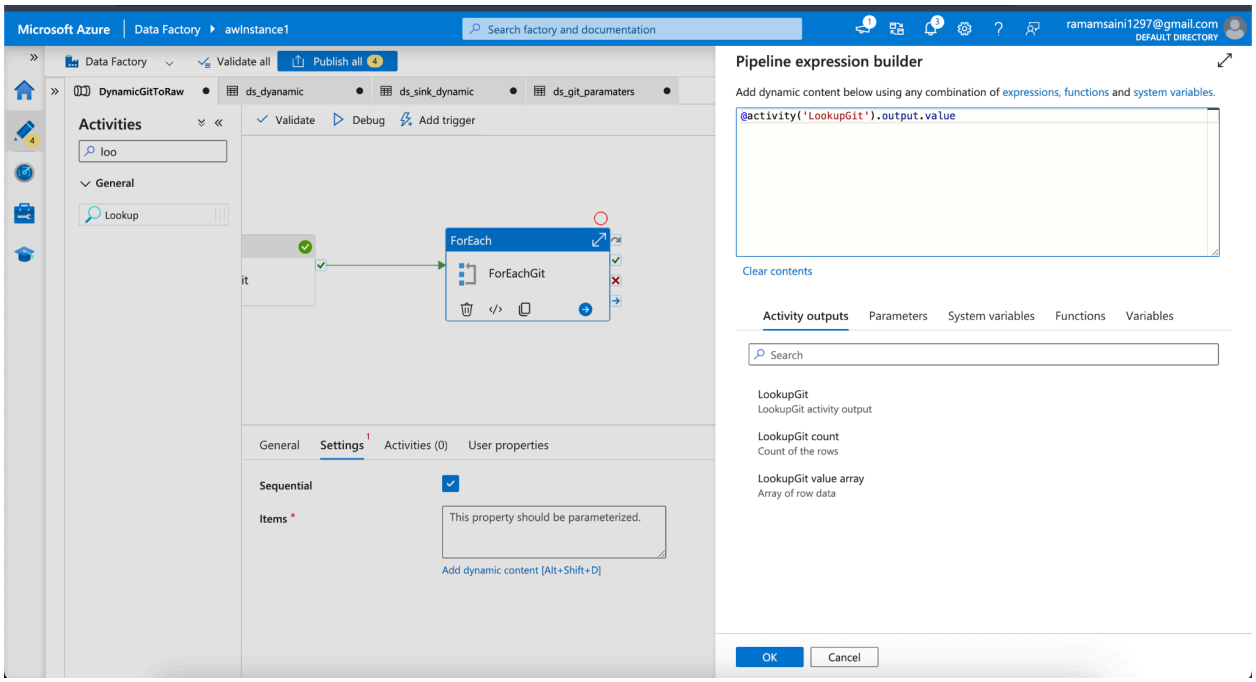
In Azure Data Factory (ADF), the Lookup activity is used to read data from a source (like a database, CSV, JSON, etc.) and make it available to other activities in the pipeline.

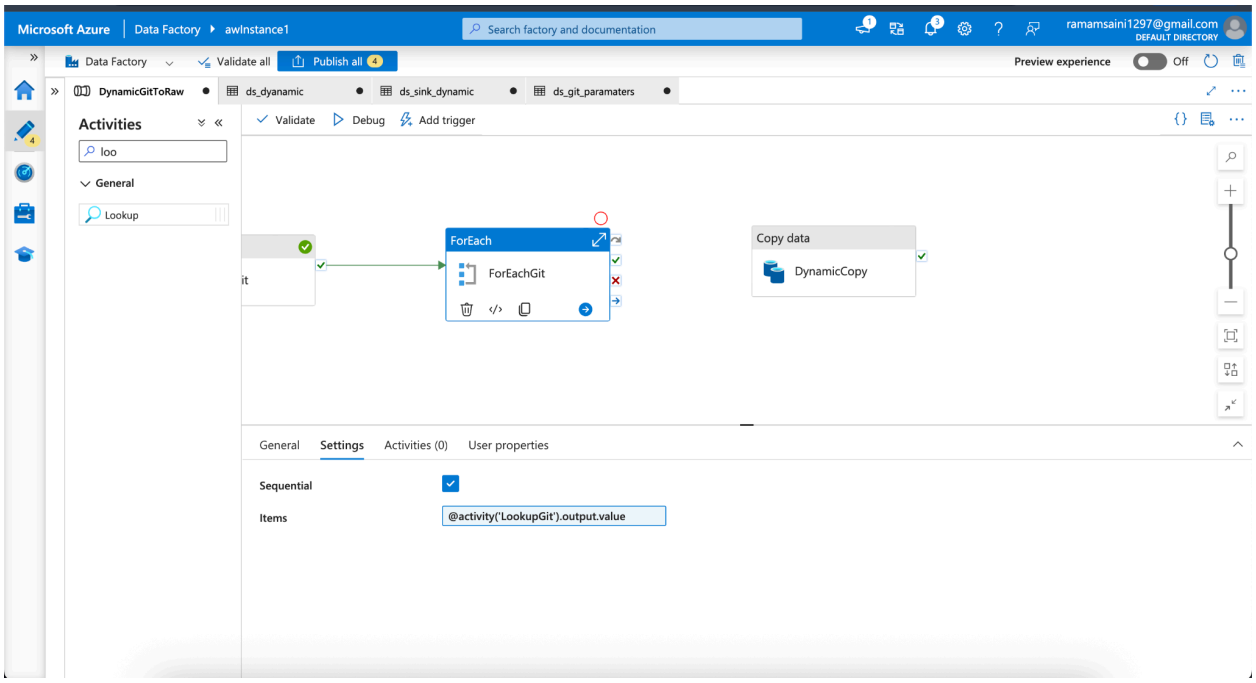
In forEach

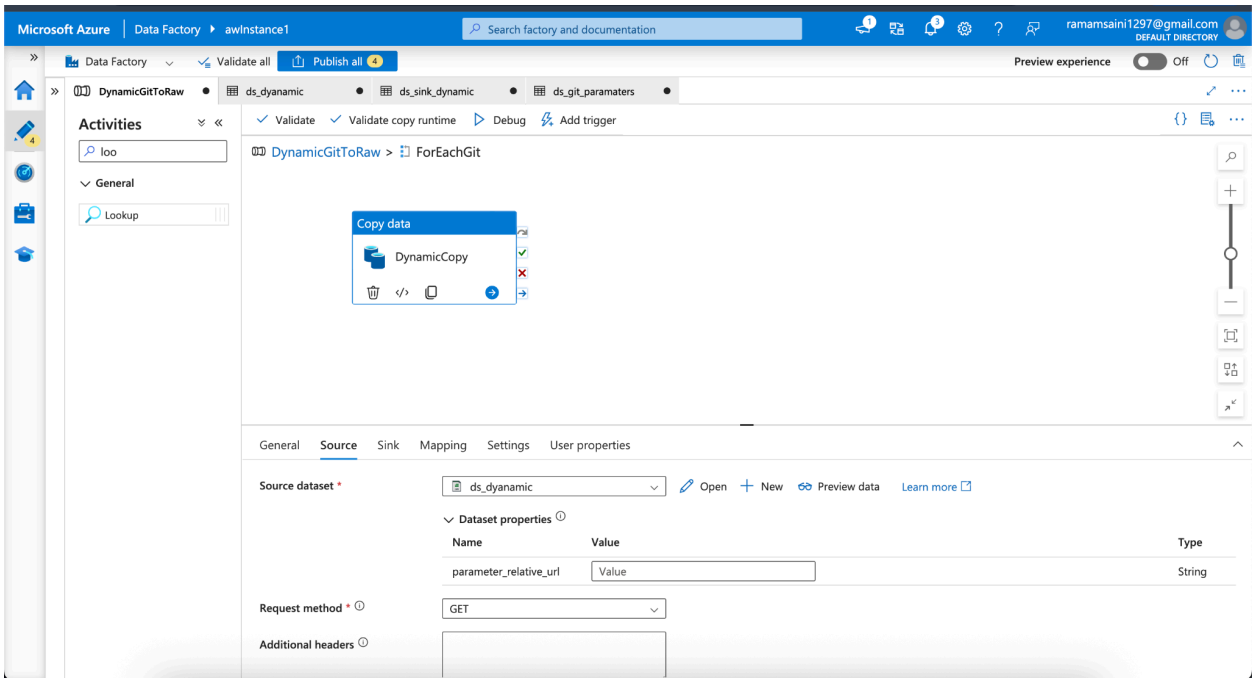


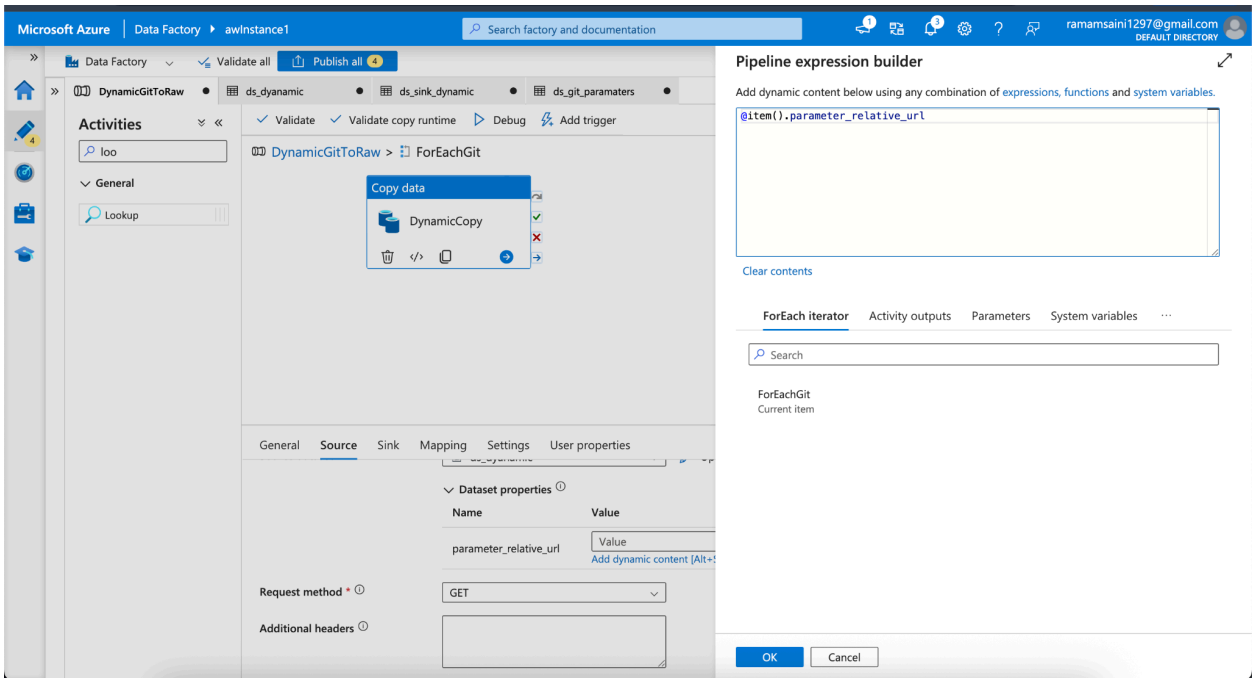
Go to settings and of foreach and paste the dynamic copy there
.value should be written .. we ve the data

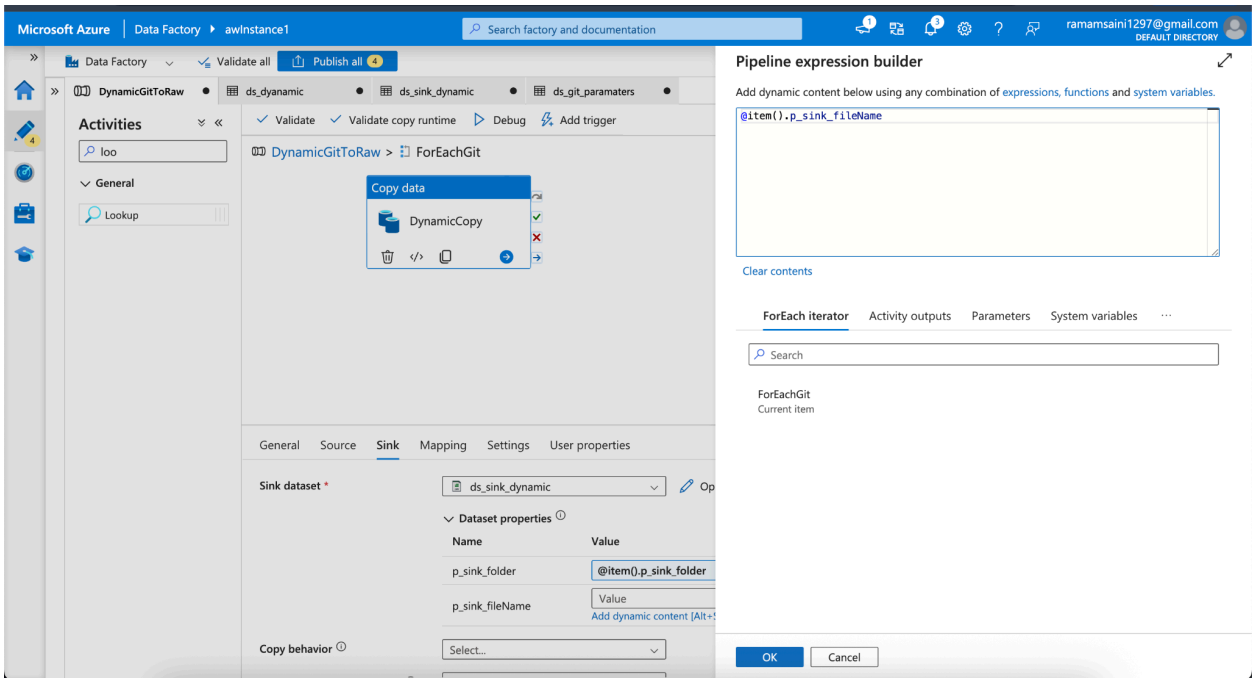


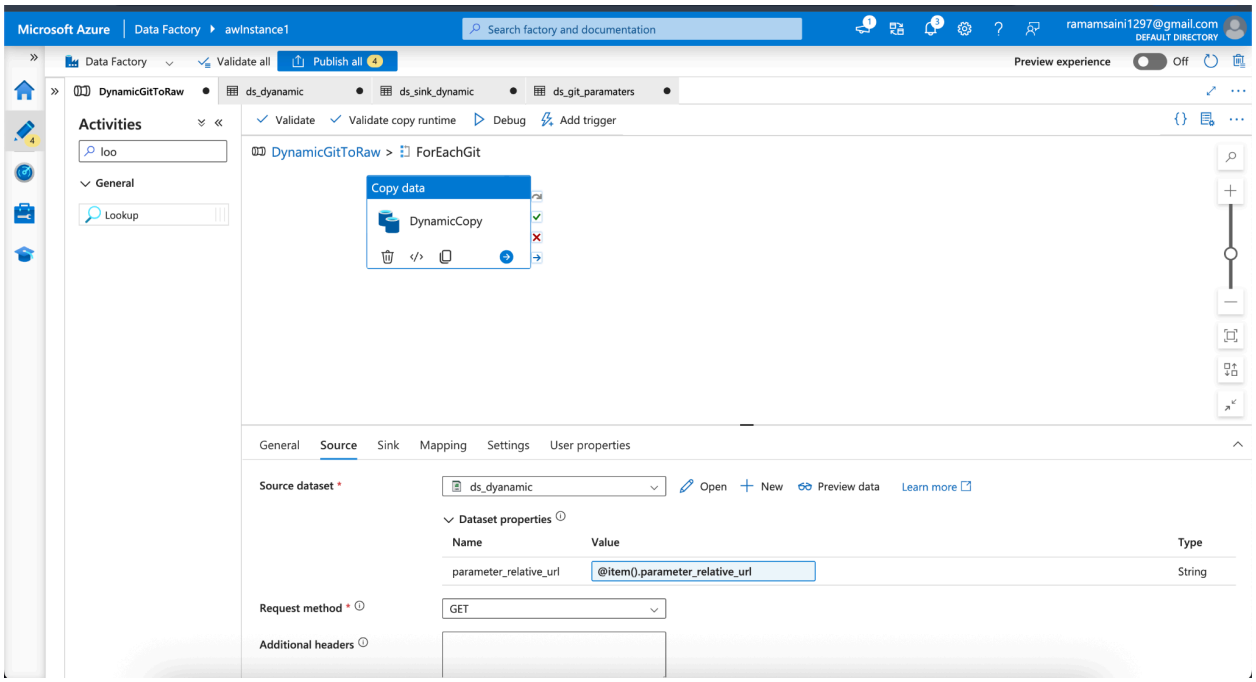

Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Data Factory
Validate all
Publish all 4
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Debug
Add trigger
ForEach
ForEachGit
General
Settings 1
Activities (0)
User properties
Sequential
Items *
This property should be parameterized.
Add dynamic content [Alt+Shift+D]
Pipeline expression builder
Add dynamic content below using any combination of expressions, functions and system variables.
@activity('LookupGit').output.value
Clear contents
Activity outputs
Parameters
System variables
Functions
Variables
Search
LookupGit
LookupGit activity output
LookupGit count
Count of the rows
LookupGit value array
Array of row data
OK
Cancel

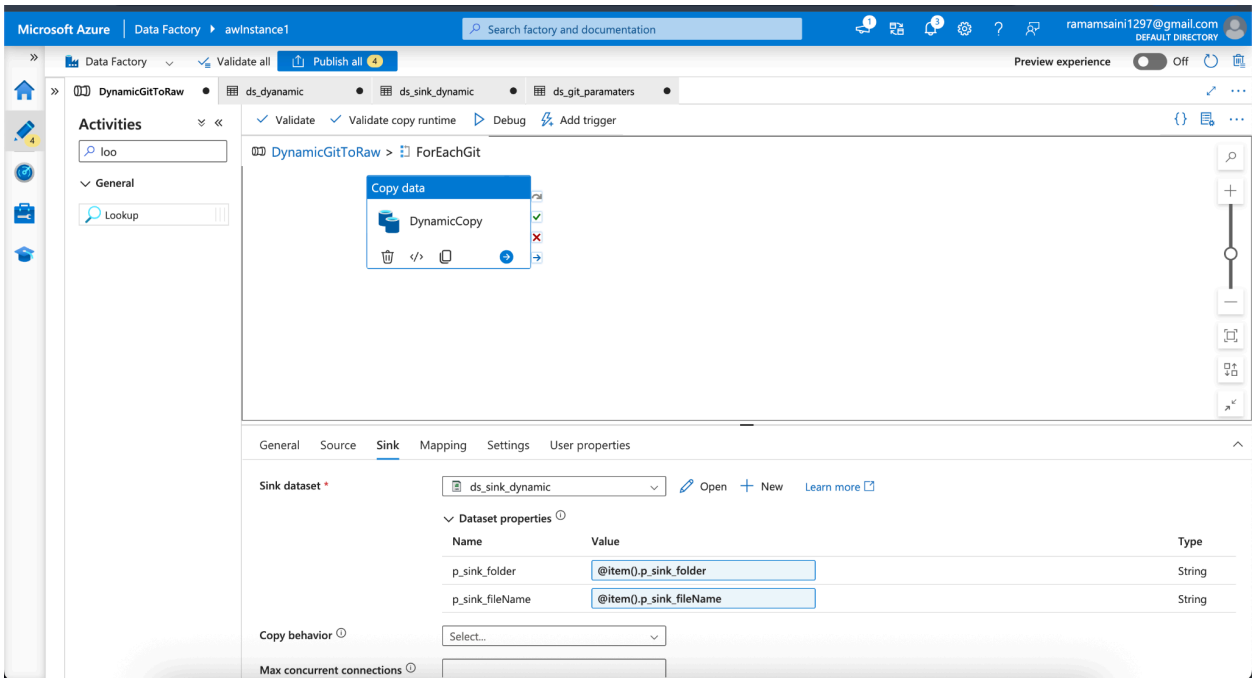

Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Data Factory
Validate all
Publish all 4
Preview experience
Off
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Debug
Add trigger
ForEach
ForEachGit
Copy data
DynamicCopy
General
Settings
Activities (0)
User properties
Sequential
Items
@activity('LookupGit').output.value


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Validate all
Publish all 4
Preview experience
Off
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Validate copy runtime
Debug
Add trigger
DynamicGitToRaw > ForEachGit
Copy data
DynamicCopy
General
Source
Sink
Mapping
Settings
User properties
Source dataset *
ds_dyanamic
Open
New
Preview data
Learn more
Dataset properties
Name
Value
Type
parameter_relative_url
Value
String
Request method *
GET
Additional headers


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Validate all
Publish all 4
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Validate copy runtime
Debug
Add trigger
DynamicGitToRaw > ForEachGit
Copy data
DynamicCopy
General
Source
Sink
Mapping
Settings
User properties
Dataset properties
Name
Value
parameter_relative_url
Value
Add dynamic content [Alt+
Request method *
GET
Additional headers
Pipeline expression builder
Add dynamic content below using any combination of expressions, functions and system variables.
@item().parameter_relative_url
Clear contents
ForEach iterator
Activity outputs
Parameters
System variables
Search
ForEachGit
Current item
OK
Cancel


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Validate all
Publish all 4
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Validate copy runtime
Debug
Add trigger
DynamicGitToRaw > ForEachGit
Copy data
DynamicCopy
Pipeline expression builder
Add dynamic content below using any combination of expressions, functions and system variables.
@item().p_sink_fileName
Clear contents
ForEach iterator
Activity outputs
Parameters
System variables
Search
ForEachGit
Current item
General
Source
Sink
Mapping
Settings
User properties
Sink dataset *
ds_sink_dynamic
Dataset properties
Name
Value
p_sink_folder
@item().p_sink_folder
p_sink_fileName
Value
Add dynamic content [Alt+
Copy behavior
Select...
OK
Cancel


Microsoft Azure
Data Factory
awInstance1
Search factory and documentation
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
Validate all
Publish all 4
Preview experience
Off
DynamicGitToRaw
ds_dyanamic
ds_sink_dynamic
ds_git_paramaters
Activities
loo
General
Lookup
Validate
Validate copy runtime
Debug
Add trigger
DynamicGitToRaw > ForEachGit
Copy data
DynamicCopy
General
Source
Sink
Mapping
Settings
User properties
Source dataset *
ds_dyanamic
Open
New
Preview data
Learn more
Dataset properties
Name
Value
Type
parameter_relative_url
@item().parameter_relative_url
String
Request method *
GET
Additional headers



AZURE DATABRICKS

3. Data Transformation with Azure Databricks

3.1 Setting up Databricks

Azure Databricks is a managed Apache Spark service for big data analytics.

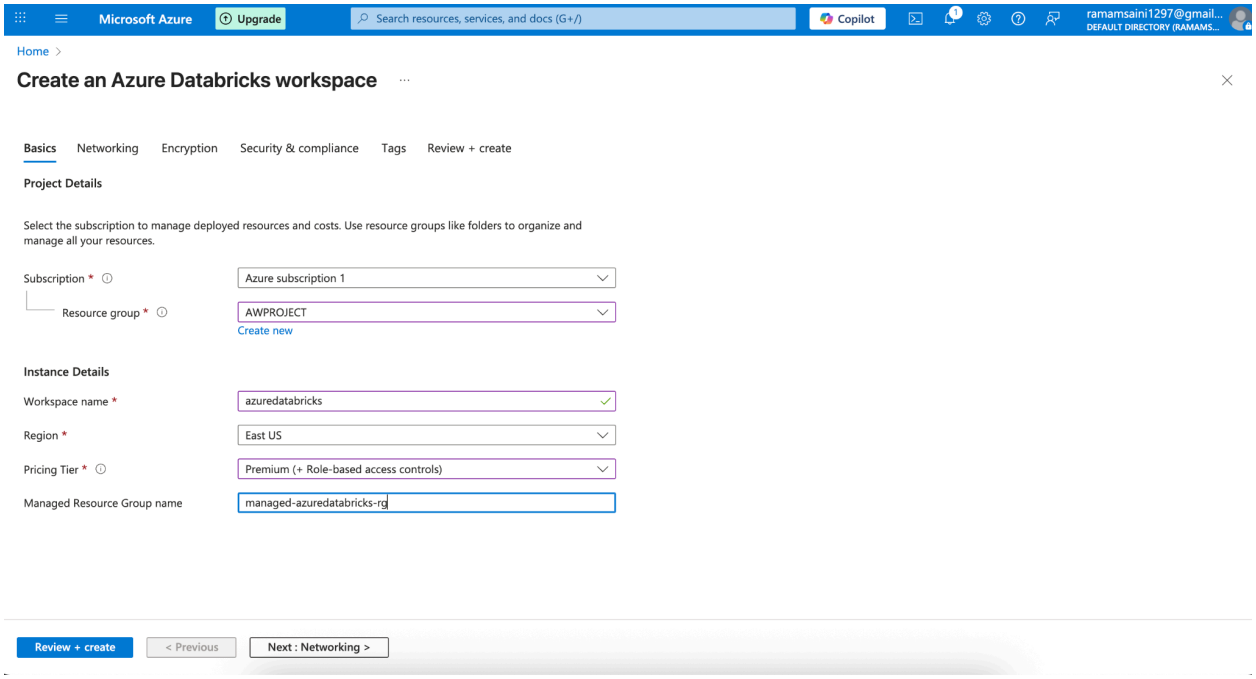
1. Create an Azure Databricks workspace in the Azure portal. A managed resource group will be created automatically to house the virtual machines and virtual network for the cluster.
2. Launch the workspace and create a Cluster under the Compute section.
3. To allow Databricks to access your data lake, you need to register an application in Microsoft Entra ID (formerly Azure Active Directory). This application will act as a service principal.
4. Register a new application (e.g., `awproject_app`), create a Client Secret for it, and save the Application (client) ID, Directory (tenant) ID, and Client Secret.
5. Assign the `awproject_app` the Storage Blob Data Contributor role on your data lake storage account to give it read/write permissions.

3.2 Performing Transformations

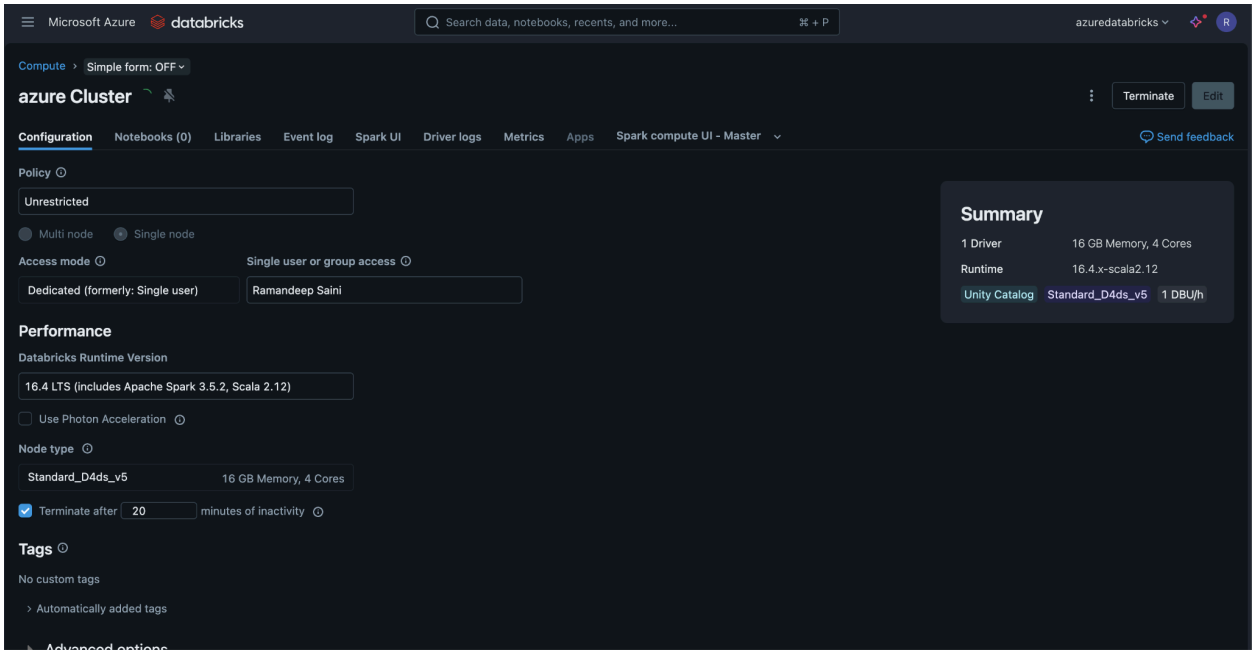
1. In your Databricks workspace, create a new Notebook.
2. Attach the notebook to your cluster.
3. Write code (e.g., using PySpark or Scala) to perform transformations. This involves reading the raw data from the `bronze` container, cleaning and enriching it, and then writing the refined data to the `silver` container.

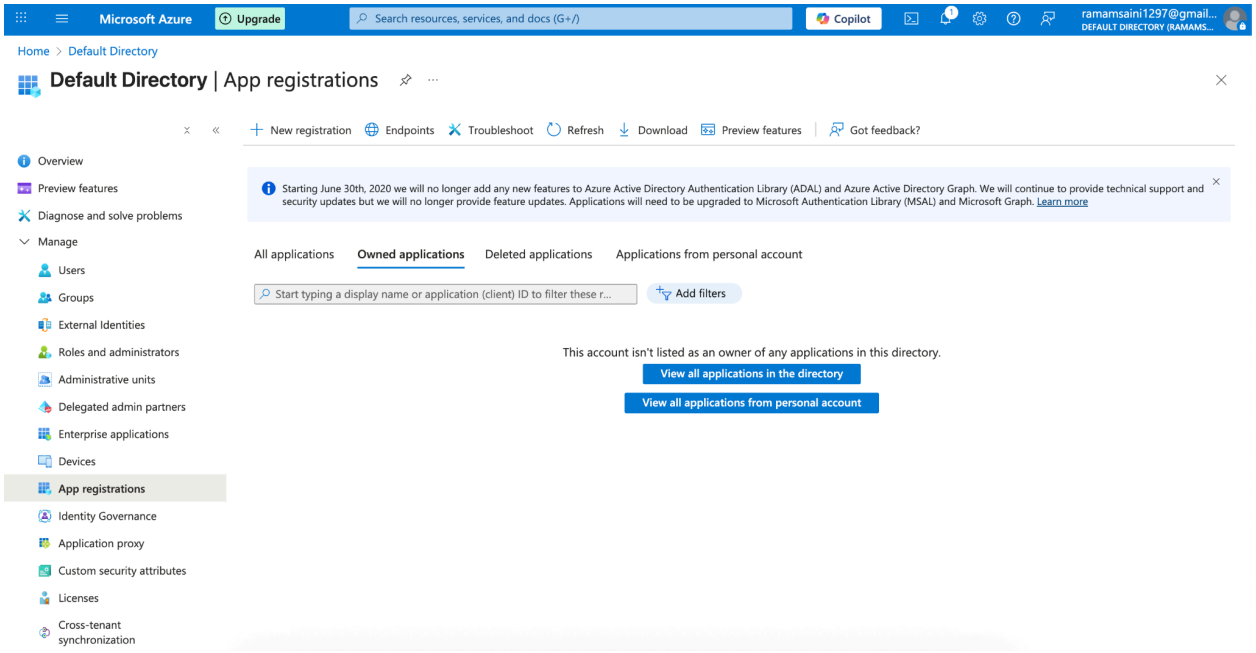
Azure databricks crete the managed resource groups where it creates all the VM / VNet

After that just create it
Go to resource —-> launch workspace

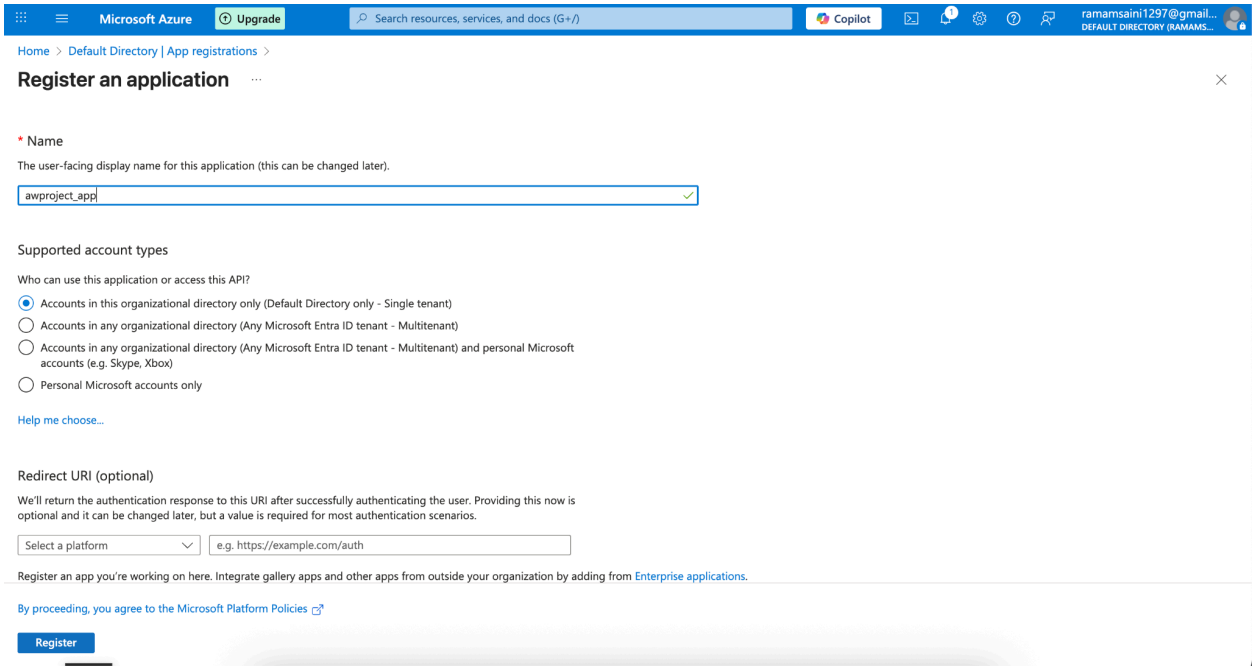


In Azure Databricks, the term "compute" usually refers to the clusters or compute resources that run your code — whether that's Spark jobs, SQL queries, ML models, or data transformations.

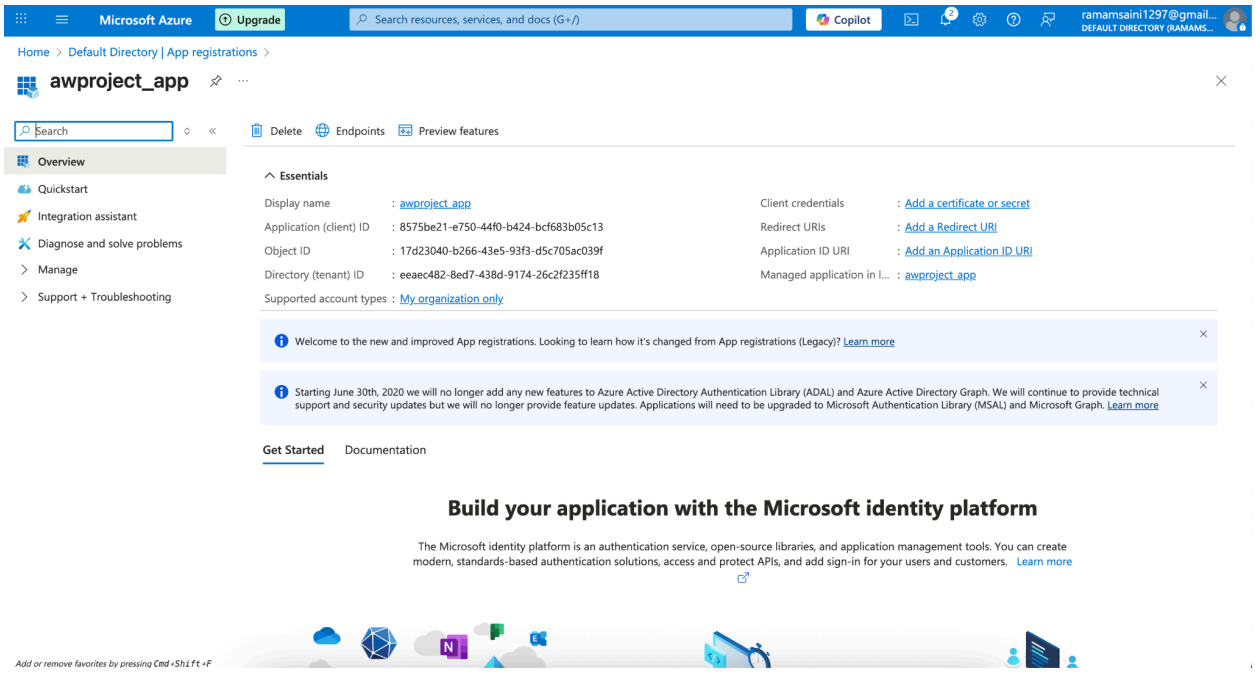




Transformation

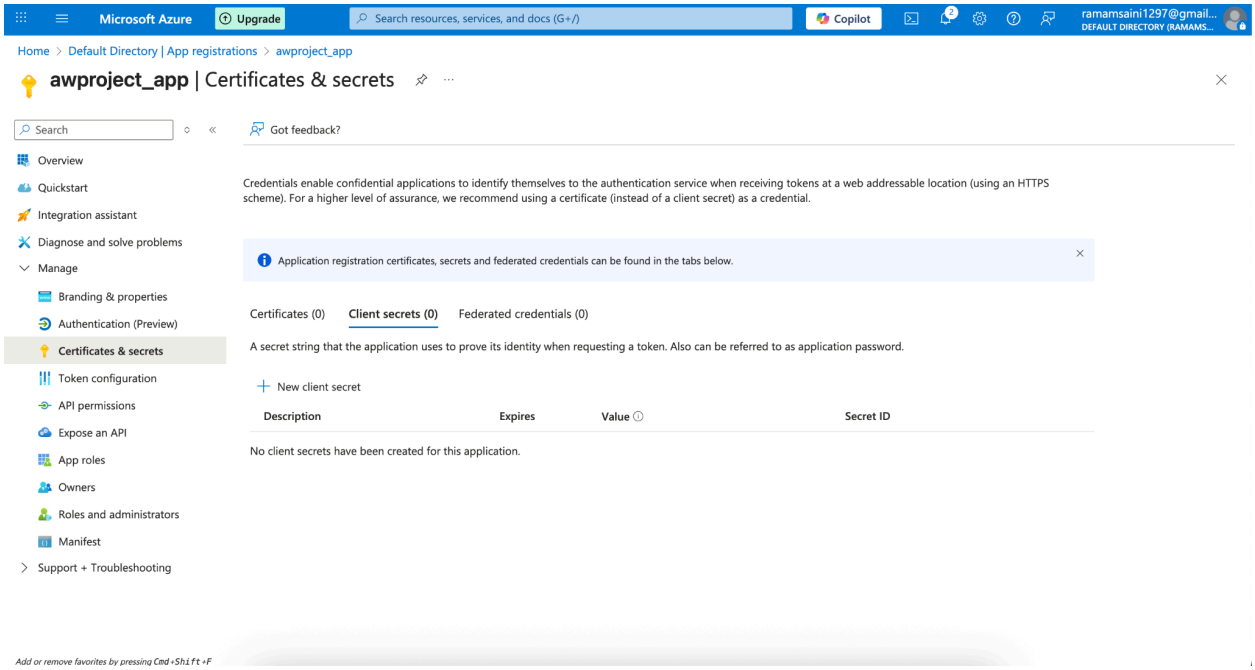


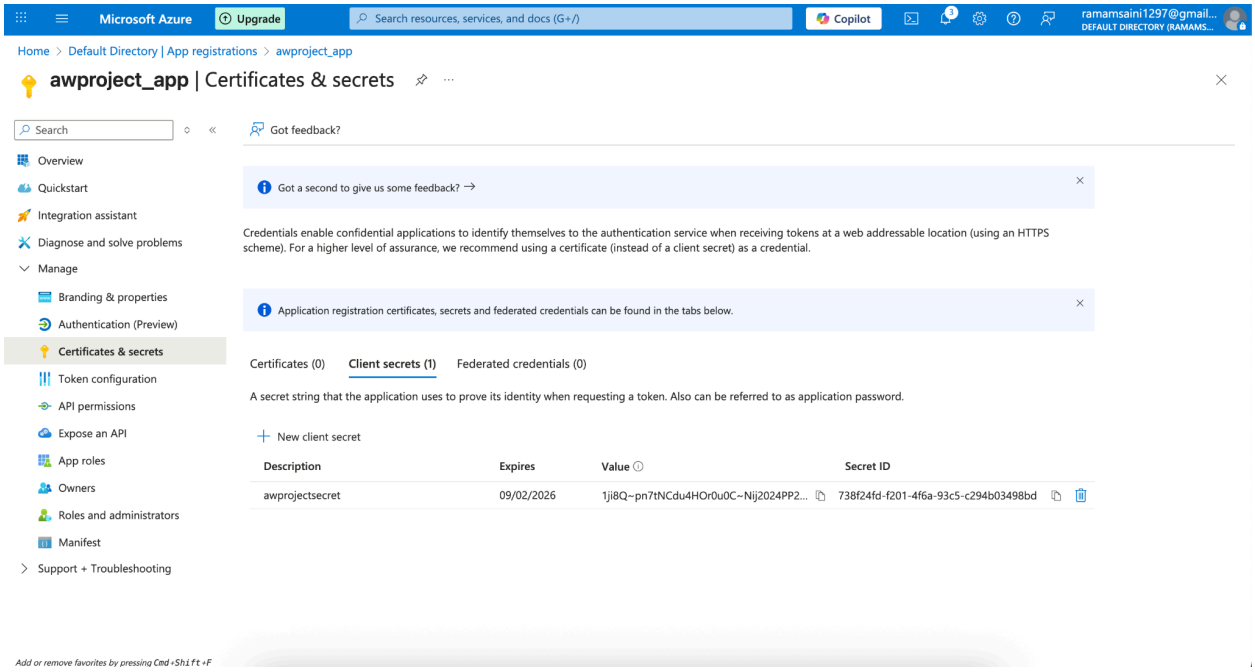
Your access to a Databricks workspace is controlled through Microsoft Entra ID.



The workspace admin grants you permissions by adding your Entra ID account to the workspace or to an Azure resource role (like *Contributor*, *Reader*, etc.).

- If you want to access data in Azure Storage without using an access key, Databricks can authenticate through Entra ID using your logged-in identity or a service principal.



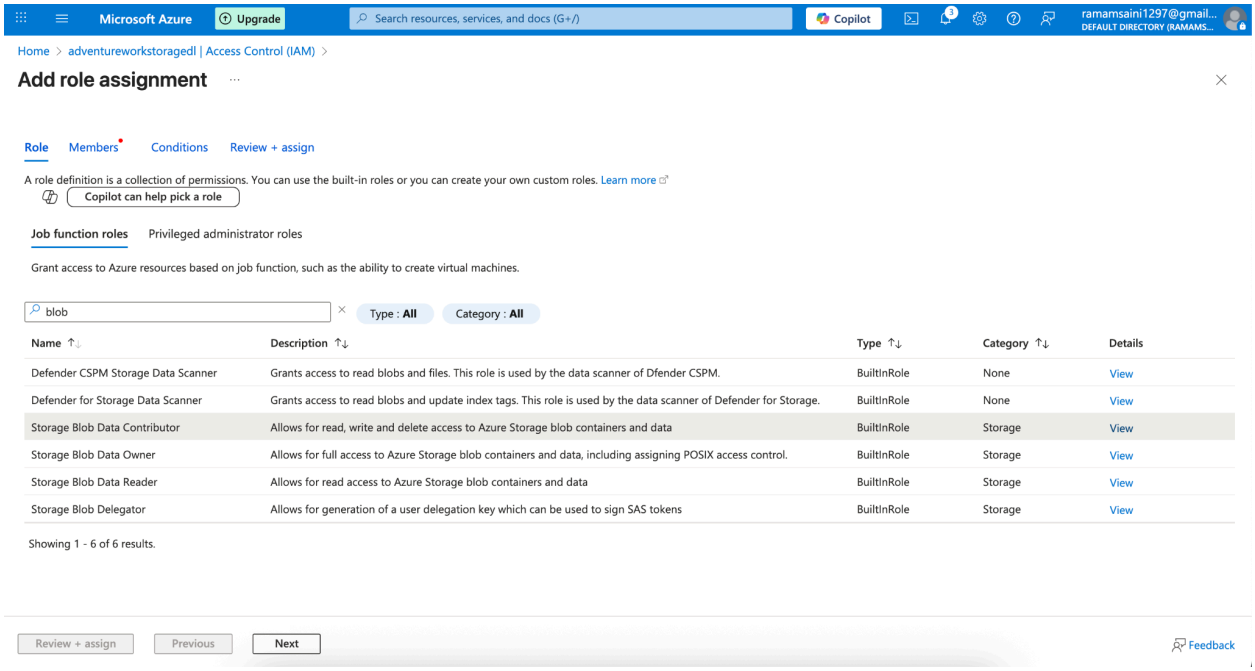


1. Creating the application

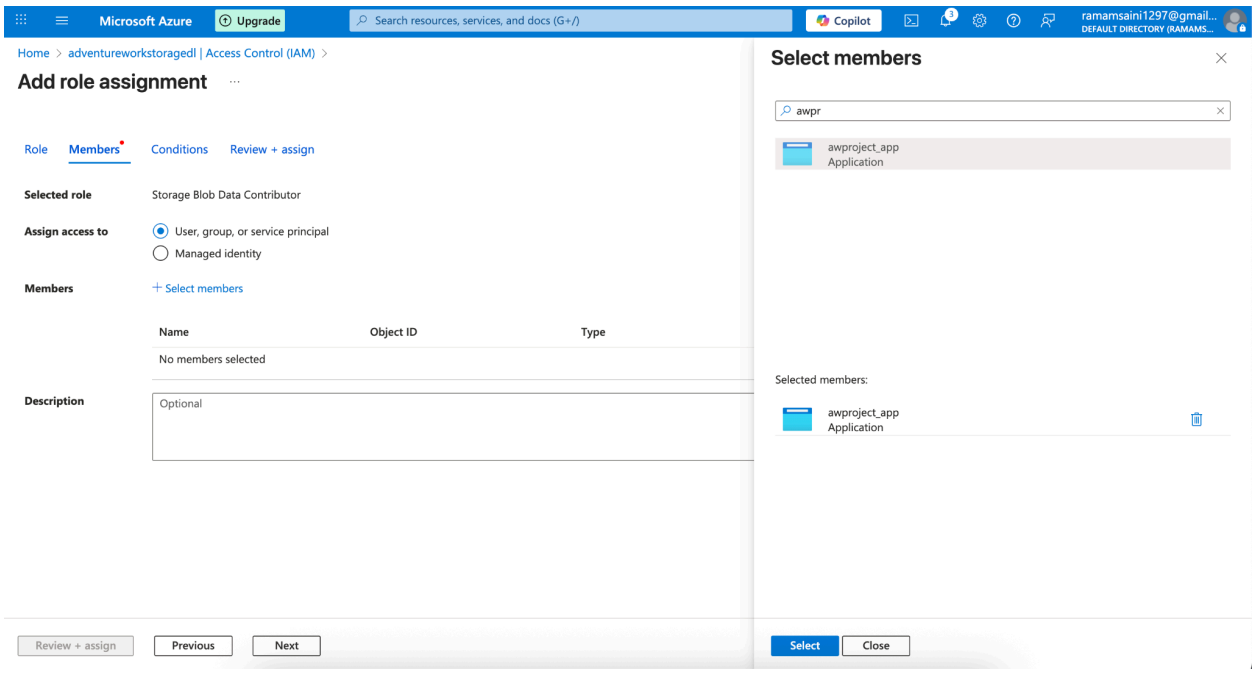
Save this info after registering application

We crete the certificate and secret

2. Assigning role to this application so it access the datalake
   Home—> storage tab→Aceess control (IAM) →Add → storage blob contributor(read_write permission)



3.



First notebook in Azure databricks

Databricks→workspace→create folder

Inside that folder→create the notebook

Transformation

Turn on our cluster →connect→select ur cluster that created in compute

Refer to notebook created

https://adb-1557609439492944.4.azuredatabricks.net/editor/notebooks/4280593522547844?o=1557609439492944#command/5655344618087748

Synapse analytics

4. Serving Data with Azure Synapse Analytics

4.1 Introduction to Synapse Analytics

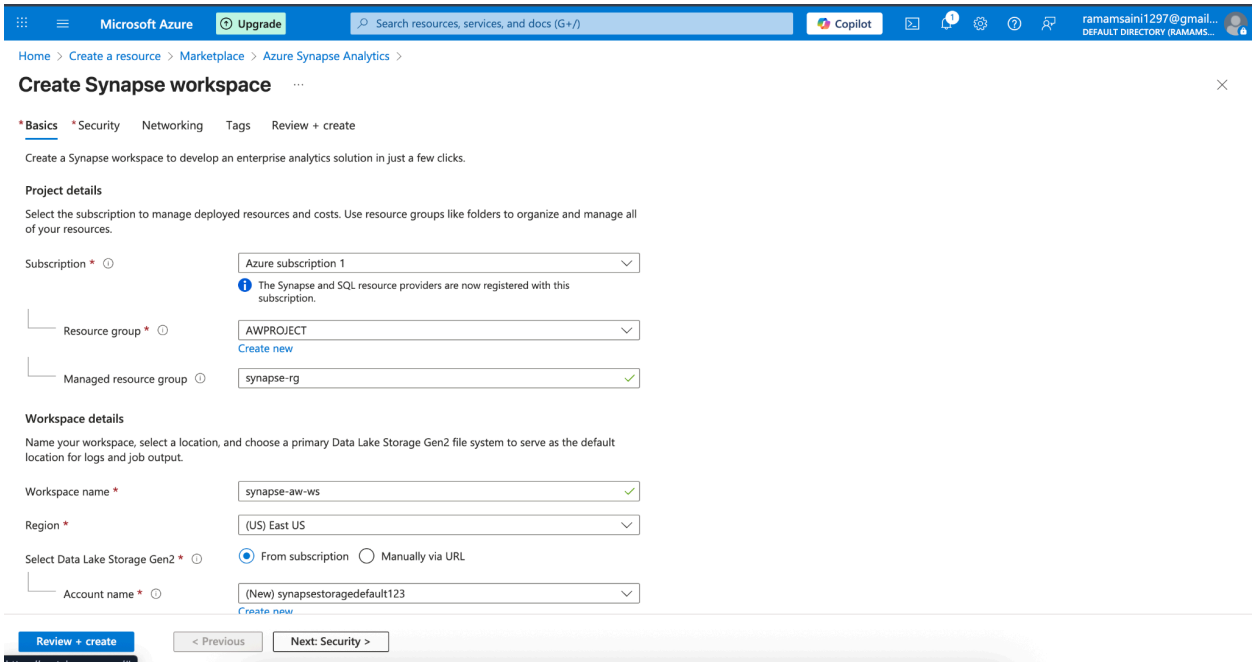
Azure Synapse Analytics is a unified platform for enterprise analytics, combining data warehousing, big data analytics, and data integration. It integrates with ADF and Spark, providing a single environment for data professionals.

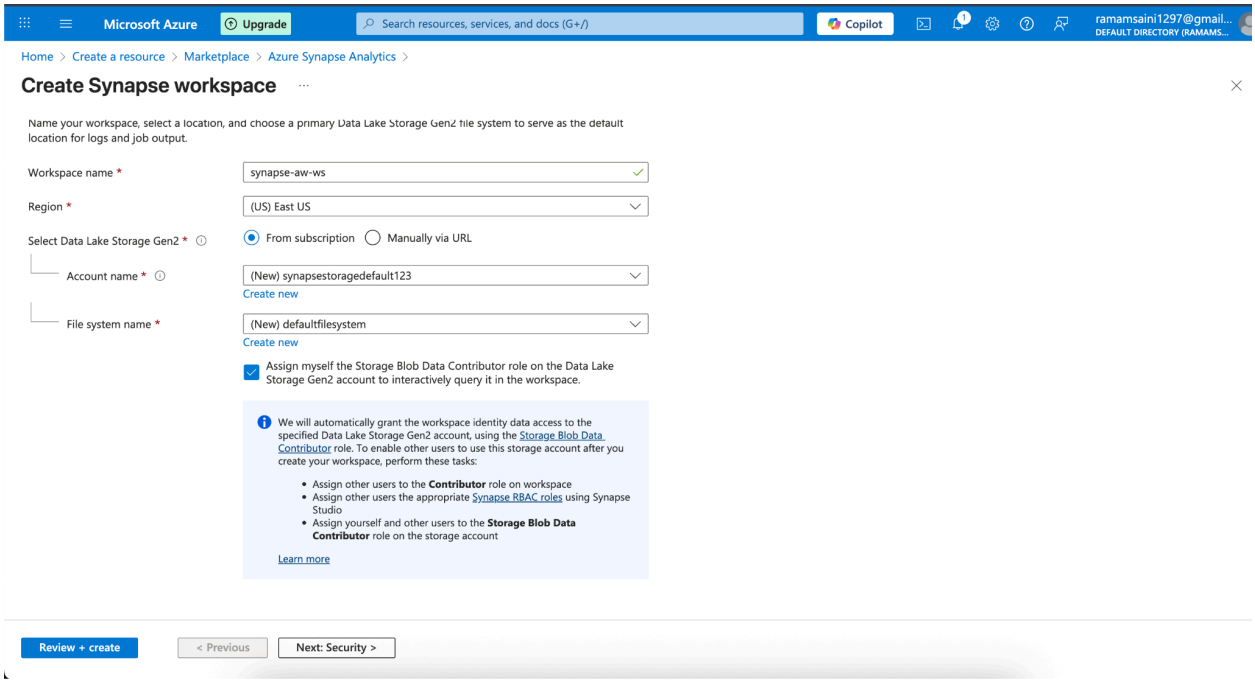
4.2 Setting up Synapse Analytics

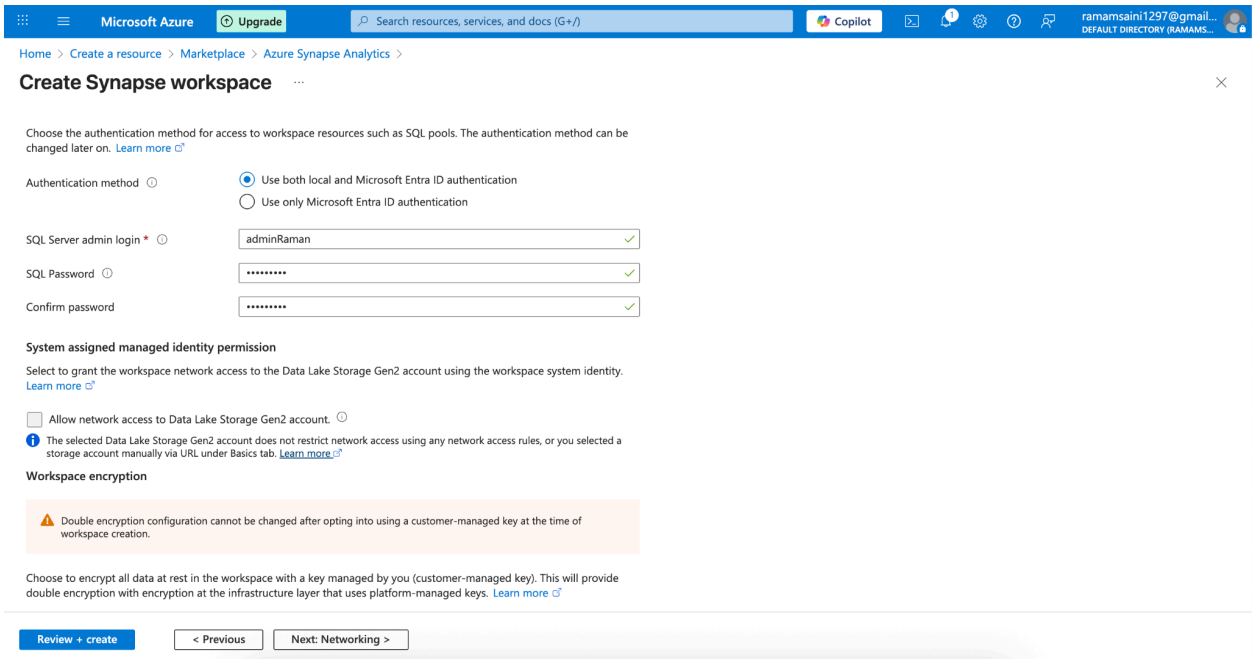
1. Create a
   Synapse workspace. You will also need to create a
   default storage account for it.
2. Assign the Synapse workspace's managed identity the
   Storage Blob Data Contributor role on your primary data lake to allow it to access the data

Whenever create synapse analytics workspace need to create the default storage account as well.

Then just create


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Copilot
ramamsaini1297@gmail...
DEFAULT DIRECTORY (RAMAMS...
Home > Create a resource > Marketplace > Azure Synapse Analytics >
Create Synapse workspace
*Basics *Security Networking Tags Review + create
Create a Synapse workspace to develop an enterprise analytics solution in just a few clicks.
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all of your resources.
Subscription *
Azure subscription 1
The Synapse and SQL resource providers are now registered with this subscription.
Resource group *
AWPROJECT
Create new
Managed resource group
synapse-rg
Workspace details
Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.
Workspace name *
synapse-aw-ws
Region *
(US) East US
Select Data Lake Storage Gen2 *
From subscription
Manually via URL
Account name *
(New) synapsestoragedefault123
Create new
Review + create
< Previous
Next: Security >

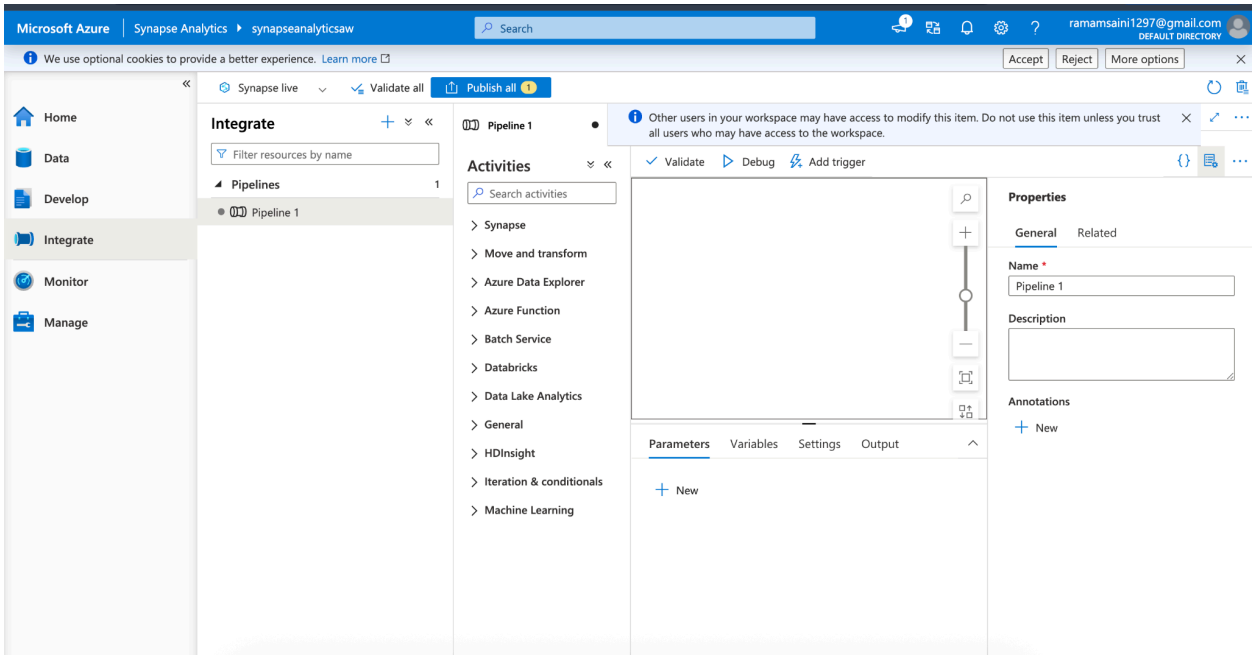

Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Copilot
ramamsaini1297@gmail...
DEFAULT DIRECTORY (RAMAMS...
Home > Create a resource > Marketplace > Azure Synapse Analytics >
Create Synapse workspace
Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.
Workspace name *
synapse-aw-ws
Region *
(US) East US
Select Data Lake Storage Gen2 *
From subscription
Manually via URL
Account name *
(New) synapsestoragedefault123
Create new
File system name *
(New) defaultfilesystem
Create new
Assign myself the Storage Blob Data Contributor role on the Data Lake Storage Gen2 account to interactively query it in the workspace.
We will automatically grant the workspace identity data access to the specified Data Lake Storage Gen2 account, using the Storage Blob Data Contributor role. To enable other users to use this storage account after you create your workspace, perform these tasks:
Assign other users to the Contributor role on workspace
Assign other users the appropriate Synapse RBAC roles using Synapse Studio
Assign yourself and other users to the Storage Blob Data Contributor role on the storage account
Learn more
Review + create
< Previous
Next: Security >



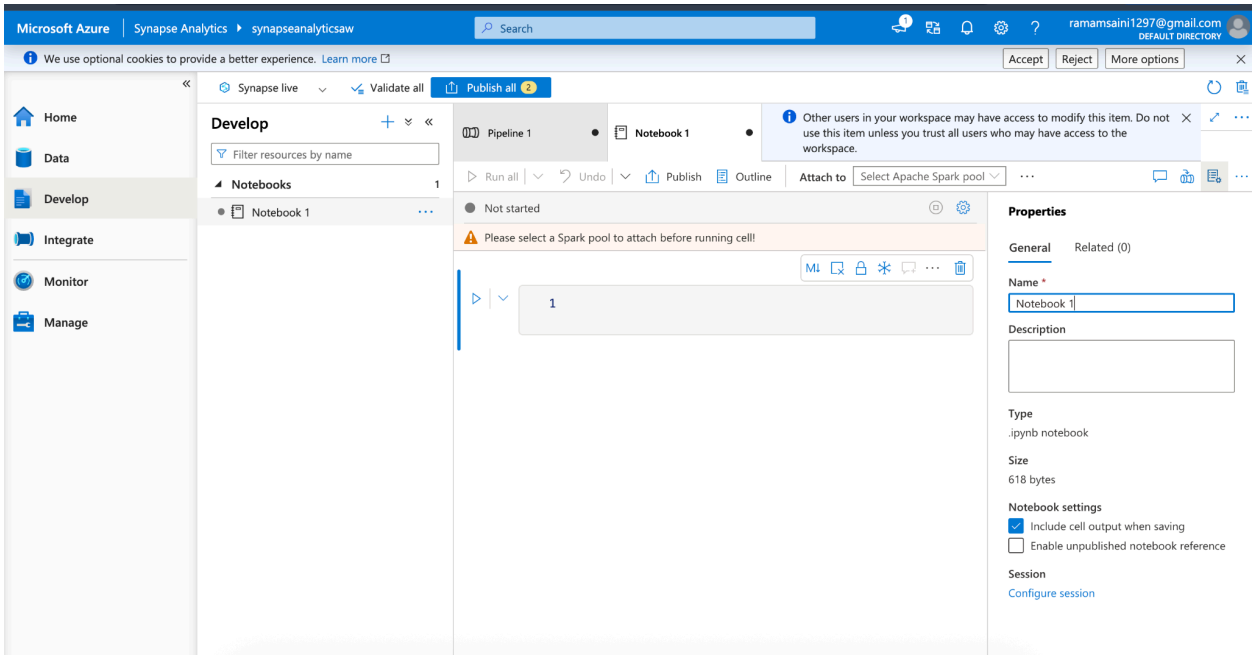
Unified platform bec we can combine

Adf + spak+warehousing

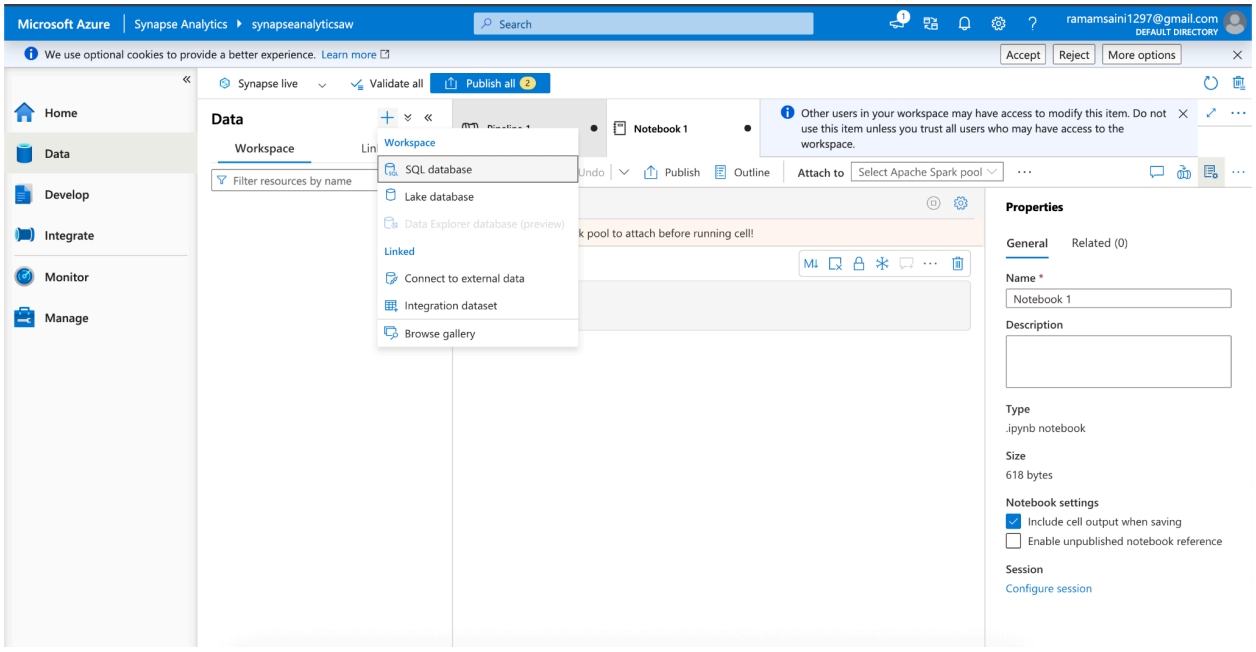
Can create pipeline inside integrate section of azure



These are spark pool having same ui like databricks but it is not databricks we call it spark pool . as both uses synapse cluster



Data warehouse



4.3 The Lakehouse Concept

Synapse Analytics supports a

Lakehouse concept, which combines a data lake with a data warehouse.

- Serverless SQL Pool: This option allows you to query data directly from your data lake without storing it in a traditional database. An abstraction layer (metadata) is created on top of the data in the data lake, which is then used by the SQL pool to process queries.
- Dedicated SQL Pool: This is a traditional data warehouse where data is physically stored in the database.

Lake db

It gives ability to use sparks. In spark when create db then it is called lake house or lake db but in databricks we called as lake house.

If creating tables using spark pool then called lake db in synapse amd in databricks we called as lake house

Create the data warehouse

Synapse db synapse datalake

Botha re azure products so no need to include nay 3rd party app., not need to register any kind of app.

Directly access data lake using synapse-analytics by allowing workspace the permission. By default synapse analytics has credentialsand we just need to assign the access , assign the role to that cred

Allow synapssse workspace to access data using the identity, usig the cred that synapse workspace or any other azure resource has bydefault

IQ

Name of that identity: managed Identity or system managed identity

Add Role

Storage data lake→IAM→ select storage blob data contributor→ next

Creating sql scripts

Develop→new→ sql script

First need to crete db

Data→ new→ sql db→2options serverless sql pool and dedicated sql pool

Dedicated sql pool: traditional way of storing data where data actually resides in db.

Traditional db on cloud and optimized for query readsfor big data for data warehousing

Serverless sql pool:like lakehouse where we do not store the data in db.

Lakehouse concept

Data resides in data lake why? Bec its cheap

Having data in csv format in data lake and want and need to create the serveless db

We want to apply select statement on this file but do not want to store the data in db traditionally

Data resides in data lake and it crete the abstraction layer (metadata layer) stores the metadata like columns, headers,all info regarding data

Whenever user queries the data we write select statement from my table

It will apply this metadata the columns on the datastored in data lake

And serveless pool do all the work behind the scene

It pull the data and apply the metadata layer and return the result

That the lake house concept where we use our datalalke but at the same time we want our data to perform as data warehouse

Data warehouse + data lake = lake house

4.4 Creating the Gold Layer

1. In Synapse Analytics, go to the
   Develop section and create a new SQL Script.

2. Create a schema (e.g., `gold`) to organize your objects.
3. Use the `OPENROWSET()` function to query data from the `silver` layer in your data lake.
4. Create Views on top of the silver layer data. A view is a saved query that doesn't store data itself.
5. Use CETAS (Create External Table As Select) to create external tables in the `gold` layer. This command reads data from the silver layer and writes it as a highly curated dataset in the `gold` folder of your data lake.
6. Before creating external tables, you need to create a master key, database-scoped credentials, external data sources for the silver and gold locations, and external file formats (e.g., `PARQUET`).

Pick the data from silver layer

Using openrowset():- helps to apply abstraction layeron the data residing in data lake

As we assign role of storage blob contributor to our synapse workspace , need to assign one more to role but to urself

Whenever query data residing in data lake , we also should have the permission to access the data

datalake→ IAM→ add→ storage blob contributor→user, grp or service contributor→select ur mail id → assign

(gold layer synapse analytics code)

External tables

External tables: that keep the data

Managed table: that do not store the data databricks or data lake etc env. store the data

Steps to create the external table in synapse analytics

1. Create Credentials

2. External data source(whenever need to pick data then we need to mention the url again and again so to avoid this we crete external data source. Keep url at container level and rest of the url at location)
3. External file format (csv, json, parquet)

1. Need to create master key for this db

We created the view on the top of silver layer , we read data from silver layer and wrote in gold layer using CETAS, and it create external table over it.

CETAS(create external table as select)

Views: not store the data, it just the query when create views then it create the views then it stores the query not the data

Ext table: we store data in data lake → gold layer→folder

To establish the connection between power BI and synapse workspace, we use SQL endpoints

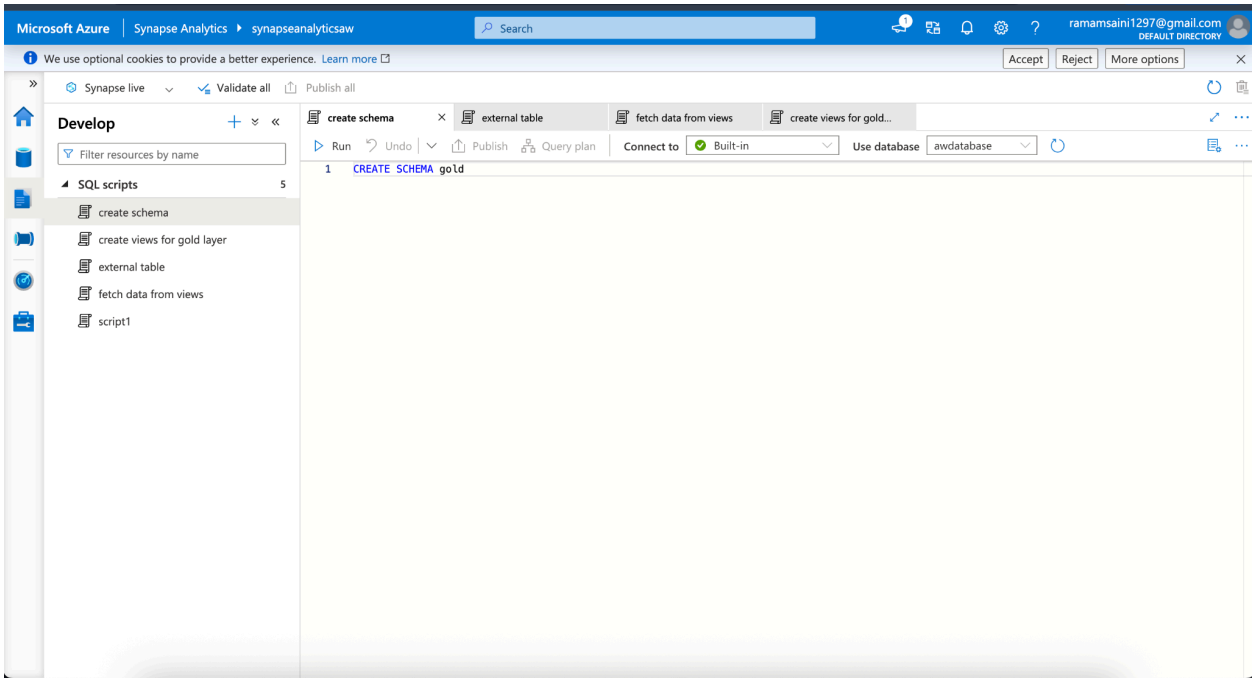
Connect to power BI

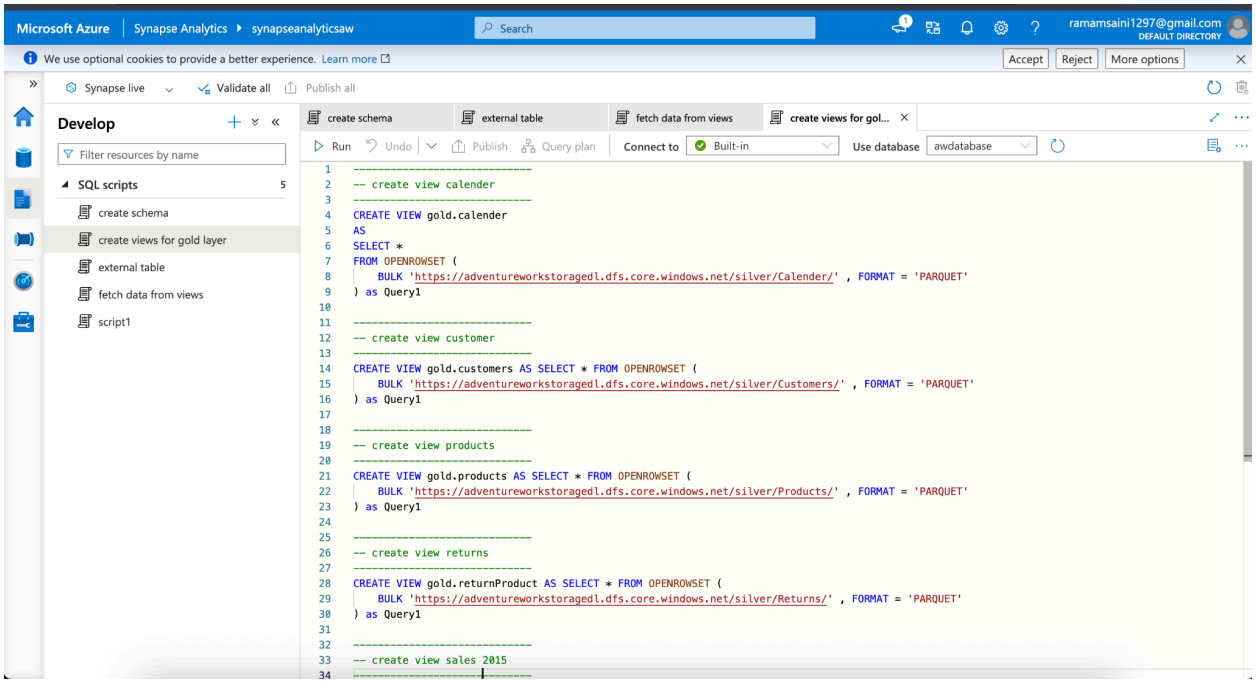
synapseAnalytics→copy serverless SQL endpoint

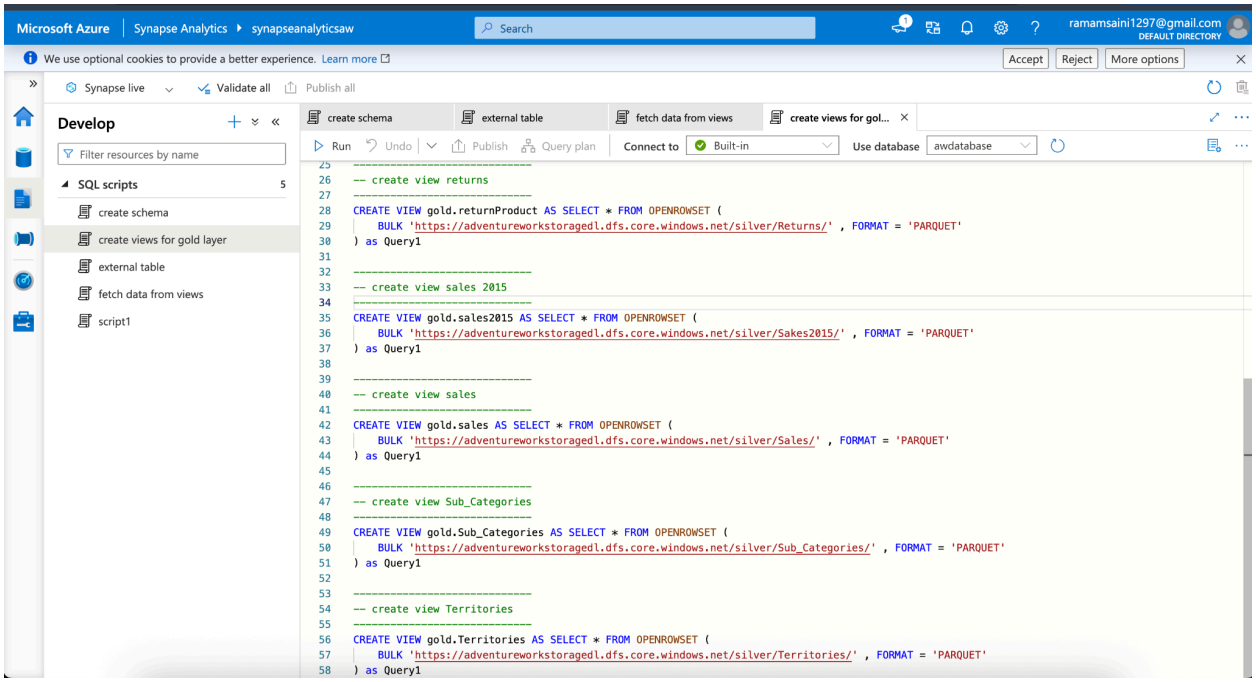
Power bi→get data→Azure→synapse analytics sql→ add sql endpoint in server

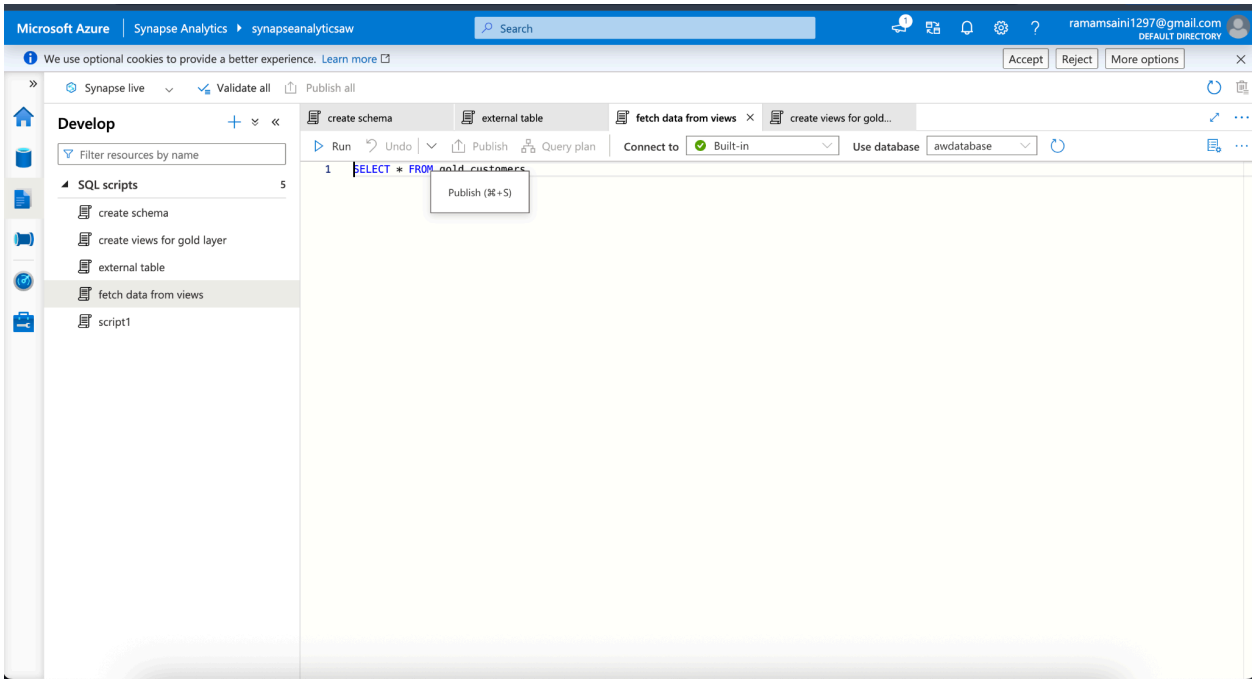
userName: adminRaman

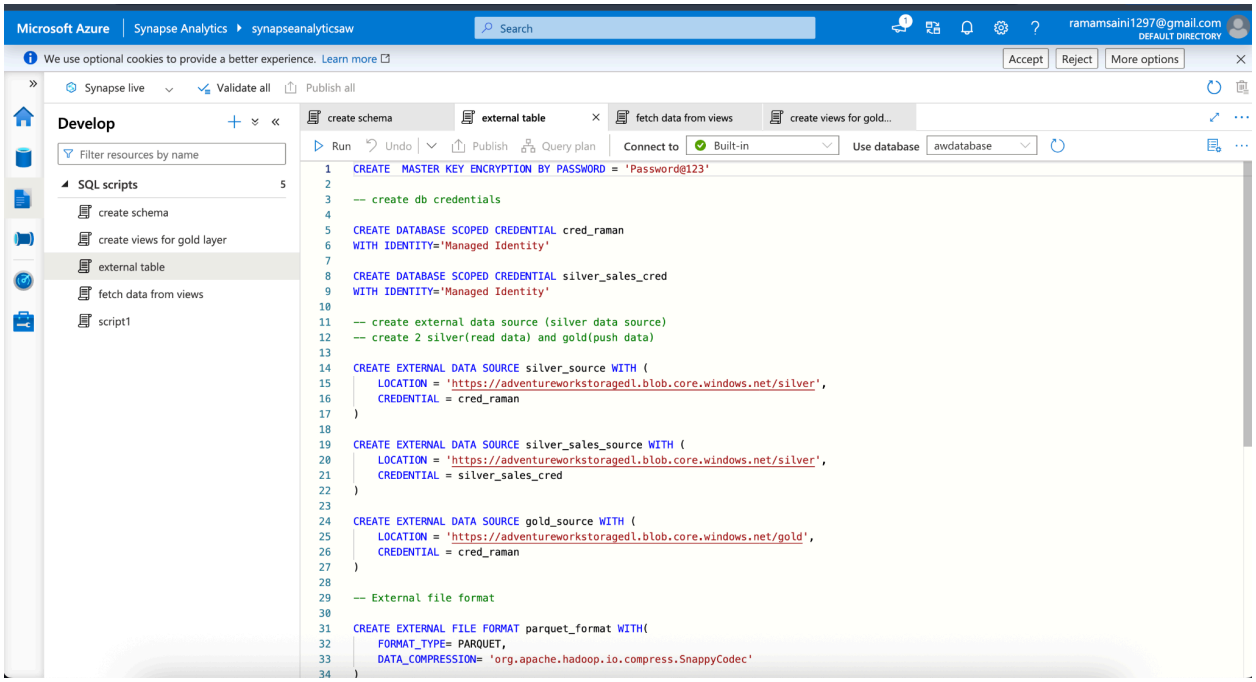
Password: Raman@!23

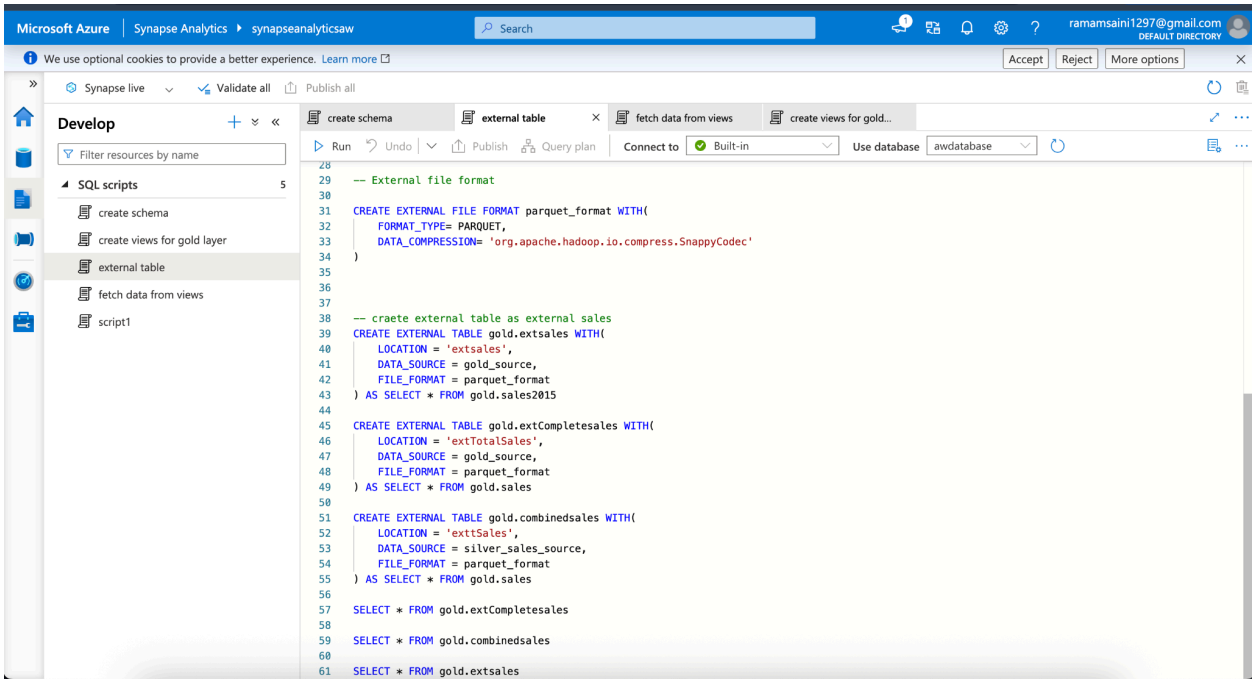

Microsoft Azure
Synapse Analytics
synapseanalyticsaw
Search
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
We use optional cookies to provide a better experience. Learn more
Accept
Reject
More options
Synapse live
Validate all
Publish all
Develop
Filter resources by name
SQL scripts 5
create schema
create views for gold layer
external table
fetch data from views
script1
create schema
external table
fetch data from views
create views for gold...
Run
Undo
Publish
Query plan
Connect to
Built-in
Use database
awdatabase
1 CREATE SCHEMA gold


Microsoft Azure
Synapse Analytics
synapseanalyticsaw
Search
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
We use optional cookies to provide a better experience. Learn more
Accept
Reject
More options
Synapse live
Validate all
Publish all
Develop
Filter resources by name
SQL scripts 5
create schema
create views for gold layer
external table
fetch data from views
script1
create schema
external table
fetch data from views
create views for gol...
Run
Undo
Publish
Query plan
Connect to
Built-in
Use database
awdatabase
1 ----------
2 -- create view calender
3 ----------
4 CREATE VIEW gold.calender
5 AS
6 SELECT *
7 FROM OPENROWSET (
8 BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Calender/' , FORMAT = 'PARQUET'
9) as Query1
10
11 ----------
12 -- create view customer
13 ----------
14 CREATE VIEW gold.customers AS SELECT * FROM OPENROWSET (
15 BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Customers/' , FORMAT = 'PARQUET'
16) as Query1
17
18 ----------
19 -- create view products
20 ----------
21 CREATE VIEW gold.products AS SELECT * FROM OPENROWSET (
22 BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Products/' , FORMAT = 'PARQUET'
23) as Query1
24
25 ----------
26 -- create view returns
27 ----------
28 CREATE VIEW gold.returnProduct AS SELECT * FROM OPENROWSET (
29 BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Returns/' , FORMAT = 'PARQUET'
30) as Query1
31
32 ----------
33 -- create view sales 2015
34 ----------


Microsoft Azure
Synapse Analytics
synapseanalyticsaw
Search
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
We use optional cookies to provide a better experience. Learn more
Accept
Reject
More options
Synapse live
Validate all
Publish all
Develop
Filter resources by name
SQL scripts 5
create schema
create views for gold layer
external table
fetch data from views
script1
create schema
external table
fetch data from views
create views for gol...
Run
Undo
Publish
Query plan
Connect to
Built-in
Use database
awdatabase
-- create view returns
CREATE VIEW gold.returnProduct AS SELECT * FROM OPENROWSET (
BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Returns/' , FORMAT = 'PARQUET'
) as Query1
-- create view sales 2015
CREATE VIEW gold.sales2015 AS SELECT * FROM OPENROWSET (
BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Sakes2015/' , FORMAT = 'PARQUET'
) as Query1
-- create view sales
CREATE VIEW gold.sales AS SELECT * FROM OPENROWSET (
BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Sales/' , FORMAT = 'PARQUET'
) as Query1
-- create view Sub_Categories
CREATE VIEW gold.Sub_Categories AS SELECT * FROM OPENROWSET (
BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Sub_Categories/' , FORMAT = 'PARQUET'
) as Query1
-- create view Territories
CREATE VIEW gold.Territories AS SELECT * FROM OPENROWSET (
BULK 'https://adventureworkstoragedl.dfs.core.windows.net/silver/Territories/' , FORMAT = 'PARQUET'
) as Query1


Microsoft Azure
Synapse Analytics
synapseanalyticsaw
Search
ramamsaini1297@gmail.com
DEFAULT DIRECTORY
We use optional cookies to provide a better experience. Learn more
Accept
Reject
More options
Synapse live
Validate all
Publish all
Develop
Filter resources by name
SQL scripts 5
create schema
create views for gold layer
external table
fetch data from views
script1
create schema
external table
fetch data from views
create views for gold...
Run
Undo
Publish
Query plan
Connect to
Built-in
Use database
awdatabase
SELECT * FROM gold.customers
Publish (⌘+S)





5. Data Visualization with Power BI

The final step is to serve the curated data to stakeholders. This can be done using

Power BI for analysis and visualization.

1. In Synapse Analytics, copy the
   serverless SQL endpoint.
2. In Power BI Desktop, select
   Get Data > Azure > Azure Synapse Analytics SQL.
3. Paste the SQL endpoint into the Server field.
4. Use the SQL server admin username and password to establish the connection.
5. You can now access the views and external tables in your gold layer to create reports and dashboards.