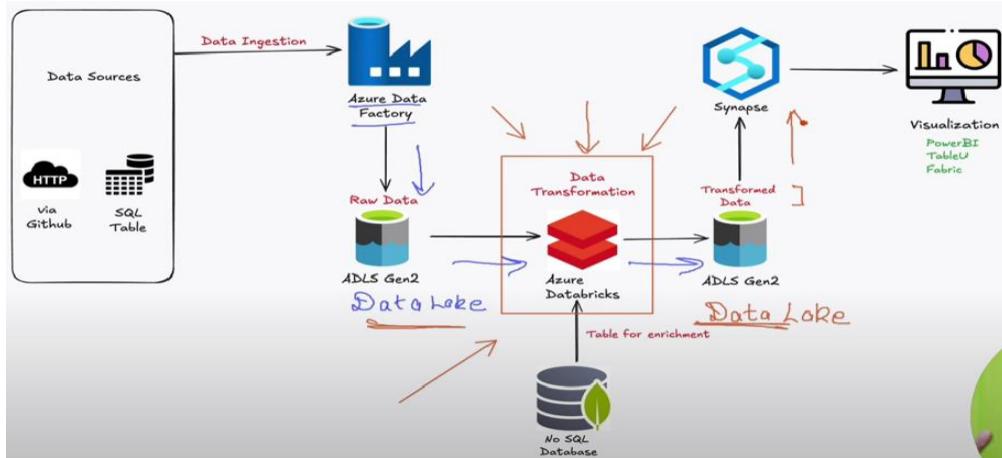
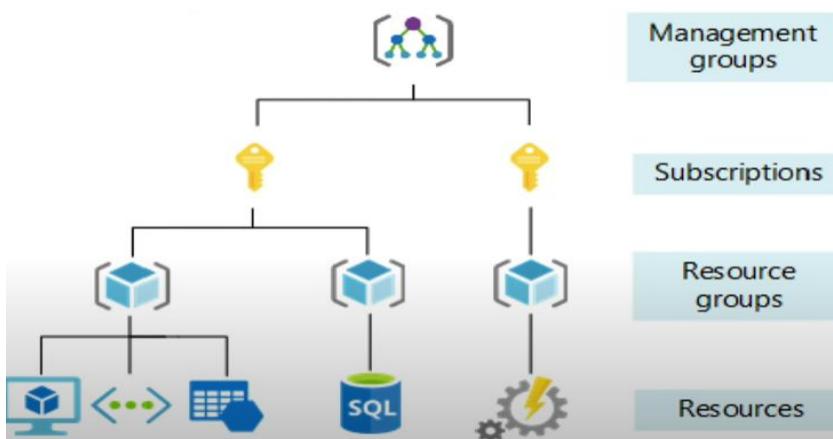


Data Engineering Project : Ecommerce

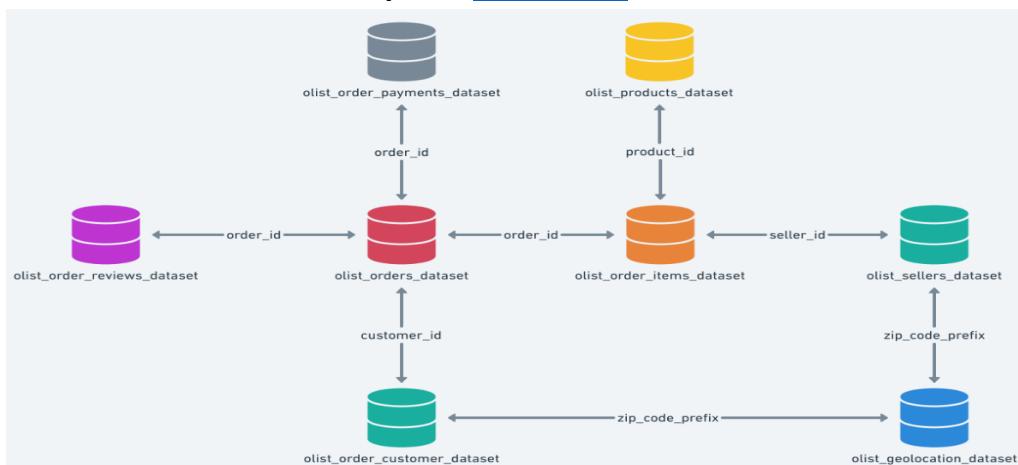
Architecture of the project



Azure account Structure



Brazilian Ecommerce Data by Olist [dataset link](#)



Github : [link](#)

Filess.io : [link](#) :-> this helps us to store our SQL and Nosql database

Data Ingestion: here we have our data in http(github) and Filess.io Database i.e we ingested here

The image shows two screenshots related to database management.

The top screenshot is from the MySQL OlistDataBases interface. It displays a database named "OlistDataBases_drivingleg" with the following details:

- Host: keuyv.h.filess.io
- Database: OlistDataBases_drivingleg
- User: OlistDataBases_drivingleg
- Port: 3307
- Password: [REDACTED]
- MySQL URI: mysql://OlistDataBases_drivingleg:...@keuyv.h.filess.io:3307/OlistDataBases_drivingleg
- MySQL login command: mysql -u OlistDataBases_drivingleg -P 3307 -p... -h keuyv.h.filess.io OlistDataBases_drivingleg

The bottom screenshot is from the Filess.io Databases list. It shows two databases:

Motor	Identifier	Available	Location	More
v8.0.29	OlistDataBases	Yes		⋮
v7.0.2	OlistMongoDb	Yes		⋮

Buttons at the bottom include "New Database", "Standard Databases", "Dedicated Databases", and "Standard VS Dedicated".

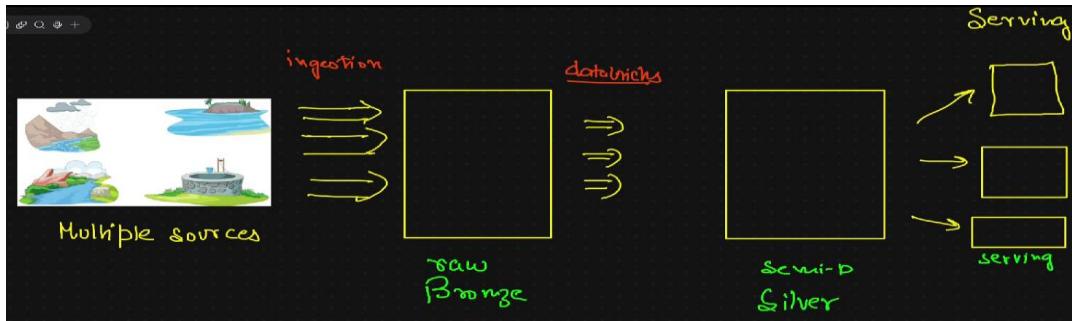
Here in above databases servers to Insert our data now form github and or through the local device

→ We just need to write a simple python scripts do insert data with MySQL-connector [link](#)

During ingestion if we face any problem of while uplading just adjust the **batch_size**

Now we have data in github and servers so perform **Ingestion** ingest this data on **Azure Data Factory**

Pipeline



Data factory

Azure Data Factory allows you to configure a Git repository with either Azure DevOps or GitHub. Git is a version control system that allows for easier change tracking and collaboration. Learn more [Set up code repository](#)

Data factory

olist-dfff

New [New](#)

Ingest Copy data at scale once or on a schedule.

Orchestrate Code-free data pipelines.

Transform data Transform your data using data flows.

Configure SSIS Manage & run your SSIS packages in the cloud.

Recent resources

No items to show

Your recently opened resources will show up here.

Ingesting data from Github

Microsoft Azure | Data Factory > olist-dfff

Factory Resources

- Pipelines
 - pipeline1
- Datasets
 - GithubData
- Data flows
- Power Query

Activities

- Copy data
- Data flow
- Synapse
- Azure Data Explorer
- Azure Function
- Batch Service
- Databricks
- Data Lake Analytics
- General
- HDIInsight
- Iteration & conditionals
- Machine Learning
- Power Query

pipeline1

Validate all

Search factory and documentation

Preview experience

Properties

General Related

Name

Description

Annotations

Annotations

Source dataset

Request method

Additional headers

Request body

Create a storage account ...

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 *

Resource group * [Create new](#)

Instance details

Storage account name *

Region * [Deploy to an Azure Extended Zone](#)

Primary service

Performance * Standard: Recommended for most scenarios (general-purpose v2 account) Premium: Recommended for scenarios that require low latency.

Redundancy *

Redundancy is the copies of this data stored in local centre if they failed or server is down you won't access the data for change it geo or zone but accordingly cost is applied higher

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

Turn on this to create folders and all in hierarchy

Properties

Data Lake Storage

Setting	Value
Hierarchical namespace	Enabled
Default access tier	Hot
Blob anonymous access	Disabled
Blob soft delete	Enabled (7 days)
Container soft delete	Enabled (7 days)
Versioning	Disabled
Change feed	Disabled
NFS v3	Disabled

Security

Setting	Value
Require secure transfer for REST API operations	Enabled
Storage account key access	Enabled
Minimum TLS version	Version 1.2
Infrastructure encryption	Disabled

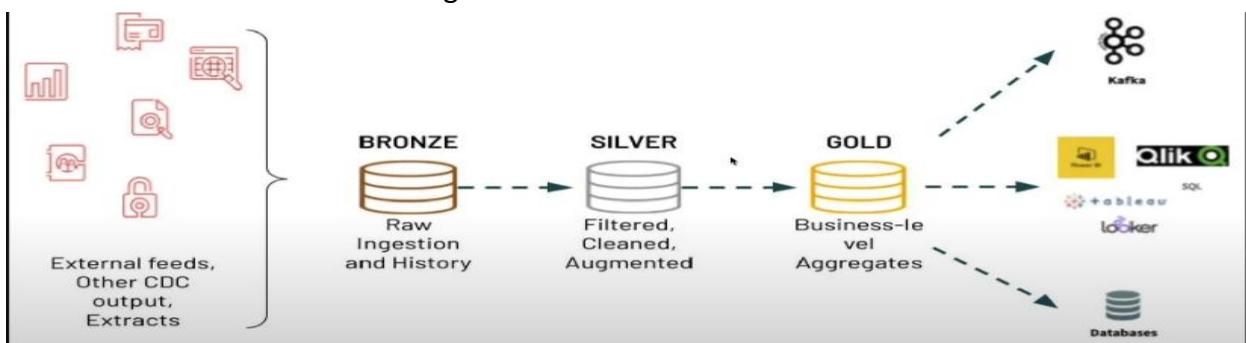
Networking

Setting	Value
Allow access from	All networks
Private endpoint connections	0

I've just created the container olistdata with directory bronze, silver, gold by medellian architec..

Name	Last modified	Access tier	Blob type	Size	Lease state
bronze/file1	7/2/2025, 1:05:15 PM				None
gold/file1	7/2/2025, 1:05:42 PM				None
silver/file1	7/2/2025, 1:05:32 PM				None

Medallian Architecture : of storing data



Now we created the container and after that created pipeline in data factory with this storage account

The screenshot shows the Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (2), and 'Data flows' (0). The main area displays 'Pipeline1' under 'Activities'. The pipeline consists of a single 'Copy data' activity named 'DataFromGitHub', which is connected to a 'DelimitedText1' dataset. The pipeline status is shown as 'Succeeded'. Below the pipeline details, a table provides a summary of the run:

Activity name	Activity st...	Activit...	Run start	Durat
DataFromGitHub	Succeeded	Copy data	7/3/2025, 12:25:45 PM	14s

Here above we need to validate and Debug this pipeline after that will get this above page tick

Here above pipeline we take data from github source and sink it to Azure Storage account with ADLS Gen2

The screenshot shows the Azure Storage Explorer interface. On the left, there's a navigation pane with 'Container' selected under 'olistdata'. The main area shows a blob named 'olist_customer' in the 'bronze' container. The blob properties are displayed on the right, including:

- URL:** https://oliststorageacc...
- LAST MODIFIED:** 7/3/2025, 12:25:57 PM
- CREATION TIME:** 7/3/2025, 12:25:57 PM
- VERSION ID:** -
- TYPE:** Block blob
- SIZE:** 8.62 MB
- ACCESS TIER:** Hot (Inferred)
- ACCESS TIER LAST MODIFIED:** N/A
- ARCHIVE STATUS:** -
- REHYDRATE PRIORITY:** -
- SERVER ENCRYPTED:** true
- ETAG:** 0x8D89FEA9A2E3C1
- VERSION-LEVEL IMMUTABILITY POLICY:** Disabled
- CACHE-CONTROL:**
- CONTENT-TYPE:** application/octet-stream
- CONTENT-MDS:**
- CONTENT-ENCODING:**
- CONTENT-LANGUAGE:**
- CONTENT-DISPOSITION:**
- LEASE STATUS:** Unlocked

Here we get that olist customer folder in storage account

tier->how frequently we can use this data ->archive , hot , etc according to cost we use this

Instead of adding single data we can use **Linked service**

->provide just base url here

The screenshot shows the Azure Data Factory 'Linked services' blade. On the left, there's a sidebar with various options like General, Connections, Source control, Author, and Security. The main area shows a 'New' button and a search bar. On the right, a 'New linked service' dialog is open for an 'HTTP' connection:

- Name:** HttpGitHubLinkService
- Description:**
- Connect via integration runtime:** AutoResolveIntegrationRuntime
- Base URL:** https://www.githubusercontent.com/mayank953/
- Server certificate validation:** Enable
- Authentication type:** Anonymous
- Auth headers:** New
- Annotations:**

At the bottom, there are 'Create', 'Back', 'Test connection', and 'Cancel' buttons. A success message 'Connection successful' is visible.

The screenshot shows the Azure Data Factory interface. On the left, the 'Linked services' blade is displayed, listing a single item: 'HttpGitHubLinkService' of type 'HTTP'. On the right, a 'New linked service' blade is open for a MySQL connection. The configuration includes:

- Name:** filessSQLDB
- Connect via integration runtime:** AutoResolveIntegrationRuntime
- Server name:** keuyv.h.filess.io
- Port:** 3307
- Database name:** OlistDataBase_drivingleg
- User name:** OlistDataBase_drivingleg
- Password:** (redacted)

A 'Create' button is at the bottom left, and a 'Connection successful' message with a 'Test connection' and 'Cancel' button is at the bottom right.

Here we linked to Database server files which mysql data

-> Will now make a pipeline of this all linked services in **Source** we have data from sources like *http* and in **Sink** we have data from *Azure Data Storage* in *bronze folder*

The screenshot shows the 'Copy data' blade in the Azure Data Factory pipeline editor. The 'Source' tab is selected. The configuration includes:

- Source dataset:** DatafromGitHubViaLinkedService
- Dataset properties:**

Name	Value
csv_relative_url	(redacted)
- Request method:** GET

Copy data

Copy Data Test

Sink

General Source Sink Mapping Settings User properties

Sink dataset *

CsvFromLinkedServiceToSink

Open New Learn more

Dataset properties

Name	Value
File_name	Value

Copy behavior

Select...

This is for each loop which helps to get multiple data in loop instead of getting single single data

Batch Service

Databricks

Data Lake Analytics

General

HDInsight

Iteration & conditionals

- Filter
- ForEach
- If Condition
- Switch
- Until
- Machine Learning

ForEach

ForEach1

Activities

No activities

Copy data

Copy Data Test

General Settings Activities (0) User properties

Name *

ForEach1

Description

This below base url we pass in for each to as we set our Relative url in main pipeline it will get all the data in a loop with iterations.

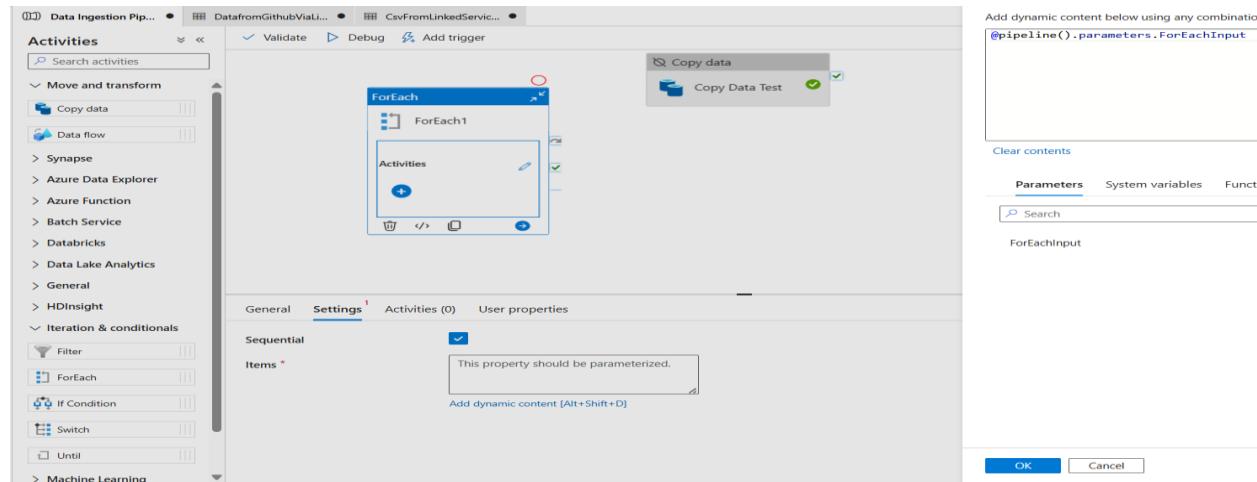
```
[  
{  
  "csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_customers_dataset.csv",  
  "file_name": "olist_customers_dataset.csv"  
},  
{  
  "csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_geolocation_dataset.csv",  
}
```

```

    "file_name": "olist_geolocation_dataset.csv"
},
{
"csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_order_items_dataset.csv",
"file_name": "olist_order_items_dataset.csv"
},
{
"csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_order_reviews_dataset.csv",
"file_name": "olist_order_reviews_dataset.csv"
},
{
"csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_orders_dataset.csv",
"file_name": "olist_orders_dataset.csv"
},
{
"csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_products_dataset.csv",
"file_name": "olist_products_dataset.csv"
},
{
"csv_relative_url": "BigDataProjects/refs/heads/main/Project-Brazilian%20Ecommerce/Data/olist_sellers_dataset.csv",
"file_name": "olist_sellers_dataset.csv"
}
]

```

We basically pasete this to our parameter “ForEachInput” which is typed Array



Pipeline expression builder

Add dynamic content below using any constructor
`@item().csv_relative_url`

ForEach iterator Parameters

Search

ForEach1 Current item

Clear contents

ForEach iterator Pa

Search

ForEach1 Current item

Clear contents

General **Source** **Sink** **Mapping** **Settings** **User properties**

Source dataset * DatafromGitHubViaLinkedService

Dataset properties

Name	Value
csv_relative_url	Value Add dynamic content [Alt+Shift+D]

General **Source** **Sink** **Mapping** **Settings** **User properties**

Sink dataset * CsvFromLinkedServiceToSink

Dataset properties

Name	Value
File_name	Value Add dynamic content [Alt+Shift+D]

Data Factory **Validate all** **Publish all 13**

Factory Resources

- Pipelines** 1
- Data Ingestion Pipeline**
- Change Data Capture (preview)** 0
- Datasets** 6
- CsvFromLinkedServiceToSink**
- DatafromGitHubViaLinkedService**
- DelimitedText1**
- GithubDataCSV**
- MySqlTable1**
- SQLToADLS**
- Data flows** 0
- Power Query** 0

Activities

- Move and transform
- Synapse
- Azure Data Explorer
- Azure Function
- Batch Service
- Databricks
- Data Lake Analytics
- General
- HDInsight
- Iteration & conditionals
- Filter
- ForEach
- If Condition
- Script

Validate **Debug** **Add trigger**

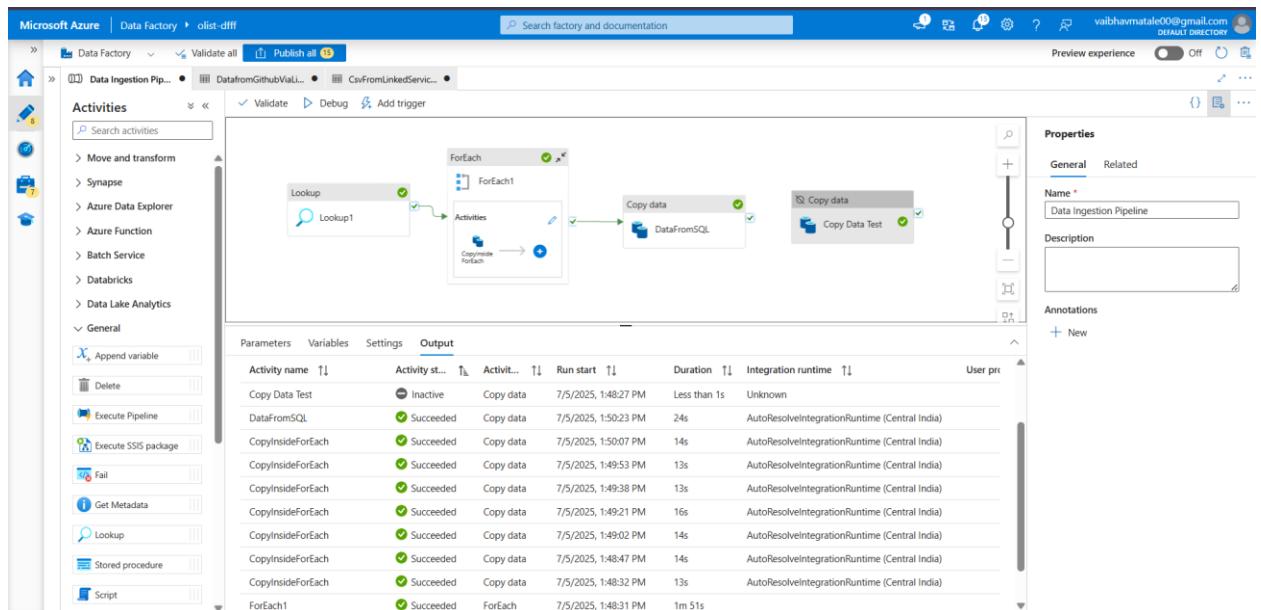
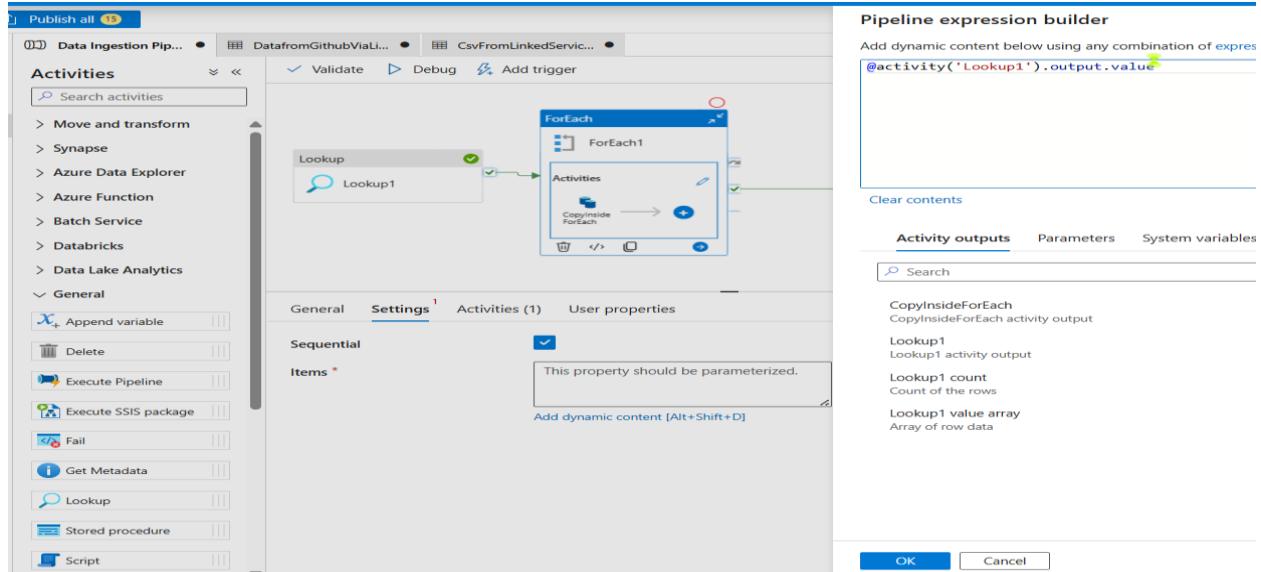
Till we have loaded data from sql server and github in our storage account

Name	Last modified	Access tier	Blob type	Size	Lease state
olist_customer	7/5/2025, 12:15:41 PM	Hot (Inferred)	Block blob	8.62 MiB	Available
olist_customers_dataset.csv	7/5/2025, 1:04:36 PM	Hot (Inferred)	Block blob	8.62 MiB	Available
olist_geolocation_dataset.csv	7/5/2025, 1:04:56 PM	Hot (Inferred)	Block blob	58.44 MiB	Available
olist_order_items_dataset.csv	7/5/2025, 1:05:12 PM	Hot (Inferred)	Block blob	14.72 MiB	Available
olist_order_payments_dataset.csv	7/5/2025, 1:06:30 PM	Hot (Inferred)	Block blob	6.28 MiB	Available
olist_order_reviews_dataset.csv	7/5/2025, 1:05:29 PM	Hot (Inferred)	Block blob	13.78 MiB	Available
olist_orders_dataset.csv	7/5/2025, 1:05:43 PM	Hot (Inferred)	Block blob	16.84 MiB	Available
olist_products_dataset.csv	7/5/2025, 1:05:57 PM	Hot (Inferred)	Block blob	2.27 MiB	Available
olist_sellers_dataset.csv	7/5/2025, 1:06:12 PM	Hot (Inferred)	Block blob	170.61 KiB	Available

Lookup

Here we pass url of links of data in json from github so we pass raw json link

Activity name	Activity state	Activity type	Run start	Duration	Integration run ID
DataFromSQL	Inactive	Copy data	7/5/2025, 1:37:06 PM	Less than 1s	Unknown
Copy Data Test	Inactive	Copy data	7/5/2025, 1:37:05 PM	Less than 1s	Unknown
ForEach1	Inactive	ForEach	7/5/2025, 1:37:05 PM	Less than 1s	AutoResolve1
Lookup1	Succeeded	Lookup	7/5/2025, 1:37:05 PM	14s	AutoResolve1



Here I published my this above pipeline

DataBricks : Databricks is a unified data analytics platform built for big data and machine learning. It is based on Apache Spark and allows you to do everything from data engineering, data science, to machine learning — all in one place.

It offers:

- A collaborative environment for data scientists, data engineers, and analysts
- Support for Python, Scala, SQL, and R

- Built-in ML & DL libraries like MLlib, TensorFlow, PyTorch, scikit-learn
- Auto-scaling and auto-termination for big data clusters
- Integration with many data sources like Azure Data Lake, AWS S3, JDBC, etc.

Azure Databricks?

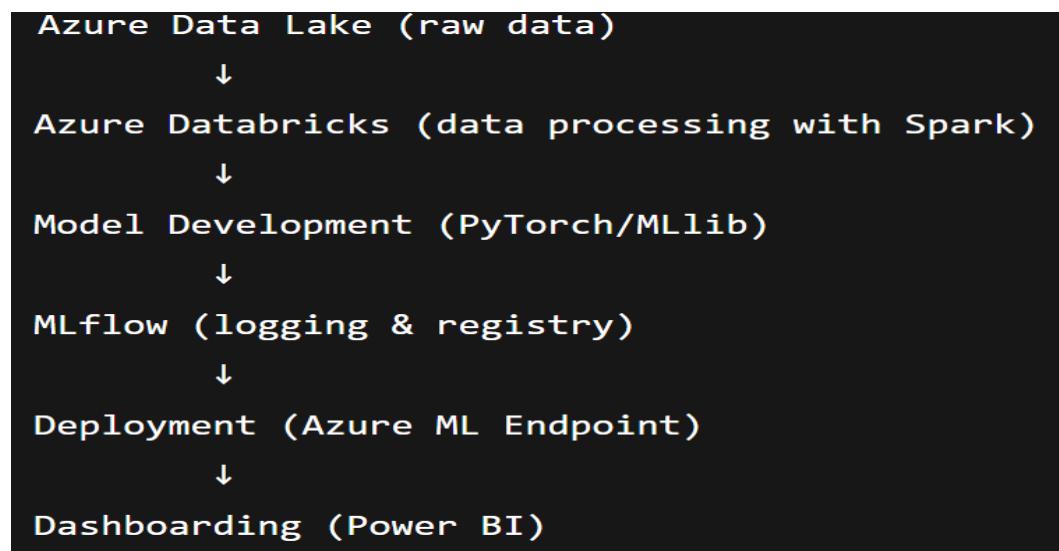
Azure Databricks is a **Microsoft Azure**-based version of Databricks, fully integrated with the Azure ecosystem.

It combines:

- The **Databricks platform**
- With **Azure services** like:
 - **Azure Data Lake Storage**
 - **Azure Synapse Analytics**
 - **Azure Machine Learning**
 - **Azure Key Vault**
 - **Azure DevOps**

Features:

- One-click setup for Spark clusters
- Enterprise-grade security with **AAD (Azure Active Directory)**
- Integrated **workspace with notebooks (like Jupyter)**
- Can connect to Power BI, Azure Blob, SQL DBs, etc



Create an Azure Databricks workspace ...

Basics Networking Encryption Security & compliance Tags Review + create

Project Details

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

Subscription * ⓘ

Azure for Students

Resource group * ⓘ

ecomm-live

[Create new](#)

Instance Details

Workspace name *

olist-spark-workshop

Region *

Central India

Pricing Tier * ⓘ

Premium (+ Role-based access controls)

i We selected the recommended pricing tier for your workspace. You can change the tier based on your needs.

Managed Resource Group name

ecomm-Databricks-resource-group

Now We do Azure DataBricks to Azure Lake Gen 2 Connections

[Refer this link For Azure Data Lake Connection Tutorial](#)

We have to do a connection for the connection to ADLS any provide app_id and other to there

Home > App registrations > olist-app-registration-DB-to-ADLS

olist-app-registration-DB-to-ADLS | Certificates & secrets

Search Got feedback?

Overview Quickstart Integration assistant Diagnose and solve problems Manage Branding & properties Authentication Certificates & secrets Token configuration API permissions Expose an API App roles Owners Roles and administrators Manifest Support + Troubleshooting

Got a second to give us some feedback? →

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (1) Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
db-client-secrete	1/3/2026	LAu8Q~Njnj... b8d6615d-9763-4c4d-bfa9-eee04acc144d	b8d6615d-9763-4c4d-bfa9-eee04acc144d

Here we pass all this in Notebook of Databricks for the connection

```

storage_account = "olistdstorageaccount"
application_id = "a87f1f454-cda5-4eb4-98fe-f41a491cb5b5"
directory_id = "af658cd9-0fe9-4706-bab4-3f452a1bdefe"

spark.conf.set("fs.azure.account.auth.type", "OAuth")
spark.conf.set("fs.azure.account.oauth.provider.type", "storage_account")
spark.conf.set("fs.azure.account.oauth2.client.provider", "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider")
spark.conf.set("fs.azure.account.oauth2.client.id", "storage_account")
spark.conf.set("fs.azure.account.oauth2.client.secret", "storage_account")
spark.conf.set("fs.azure.account.oauth2.client.endpoint", "https://login.microsoftonline.com/" + directory_id + "/oauth2/token")

```

Assigning Role to access storage containers in Azure with that we've created registration olist-db-adls

Home > olistdstorageaccount | Containers > olistdata | Access Control (IAM) > Add role assignment ...

Role Members Conditions Review + assign

Selected role Storage Blob Data Contributor

Assign access to User, group, or service principal Managed identity

Members + Select members

Name	Object ID	Type
No members selected		

Description Optional

Select members

Selected members:

olist-app-registration-DB-to-ADLS Application

