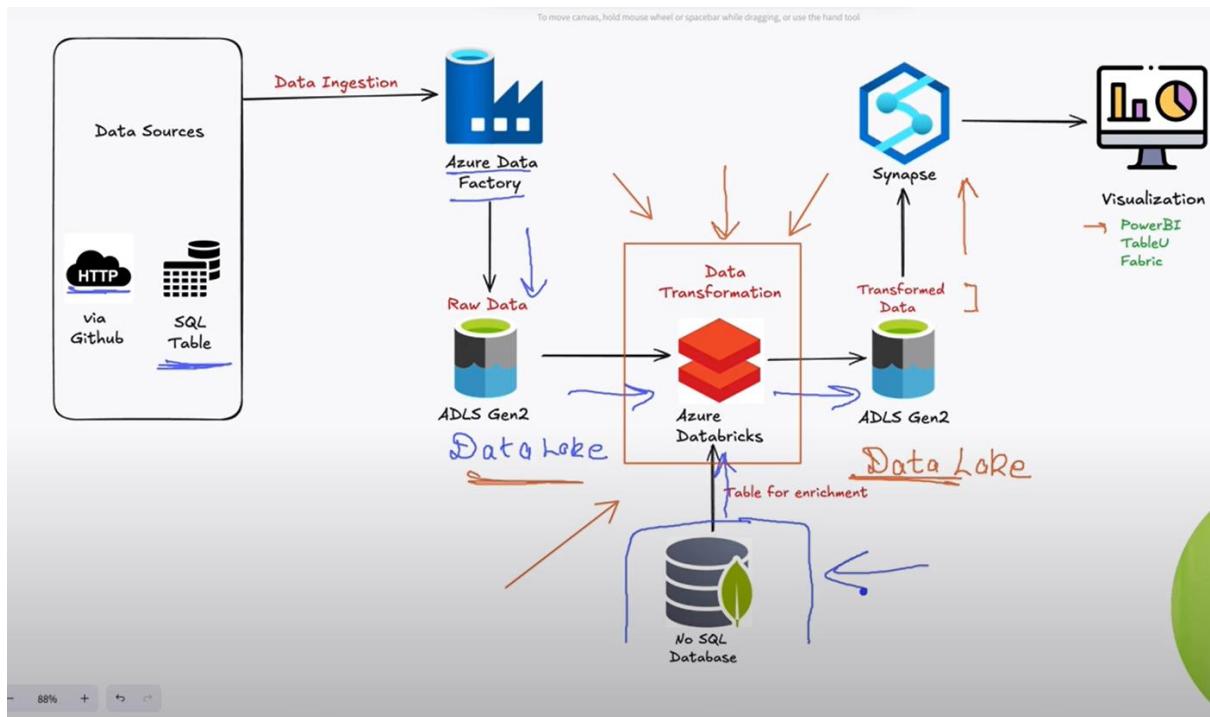
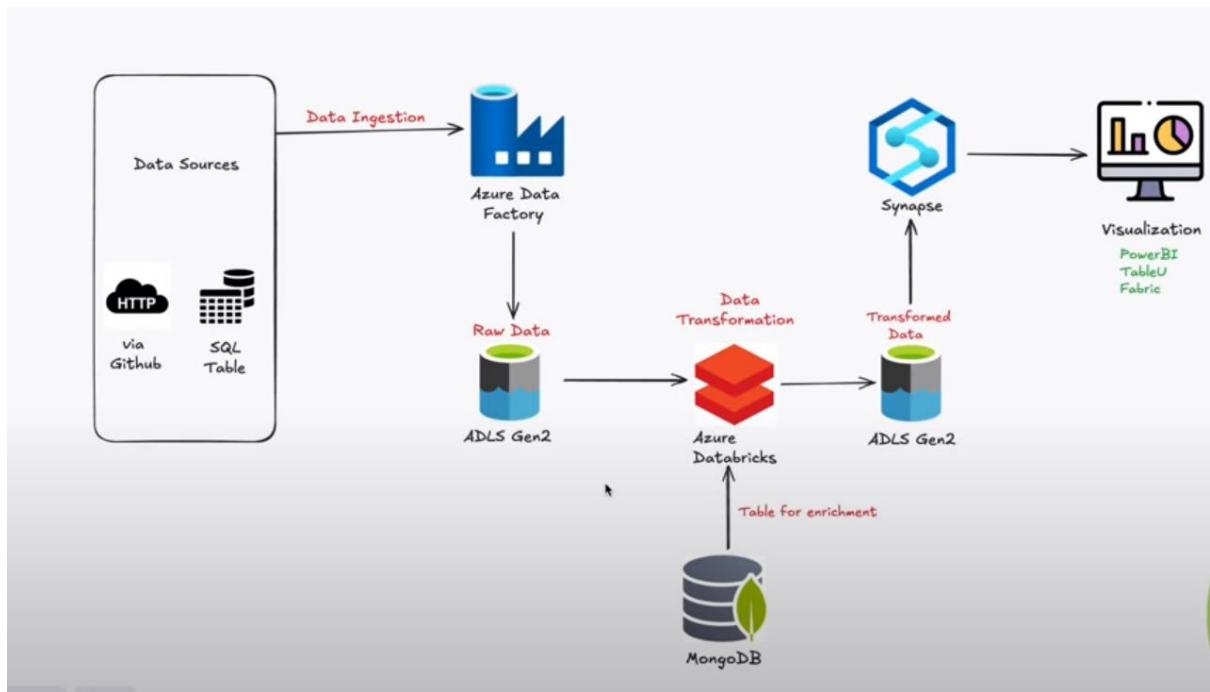
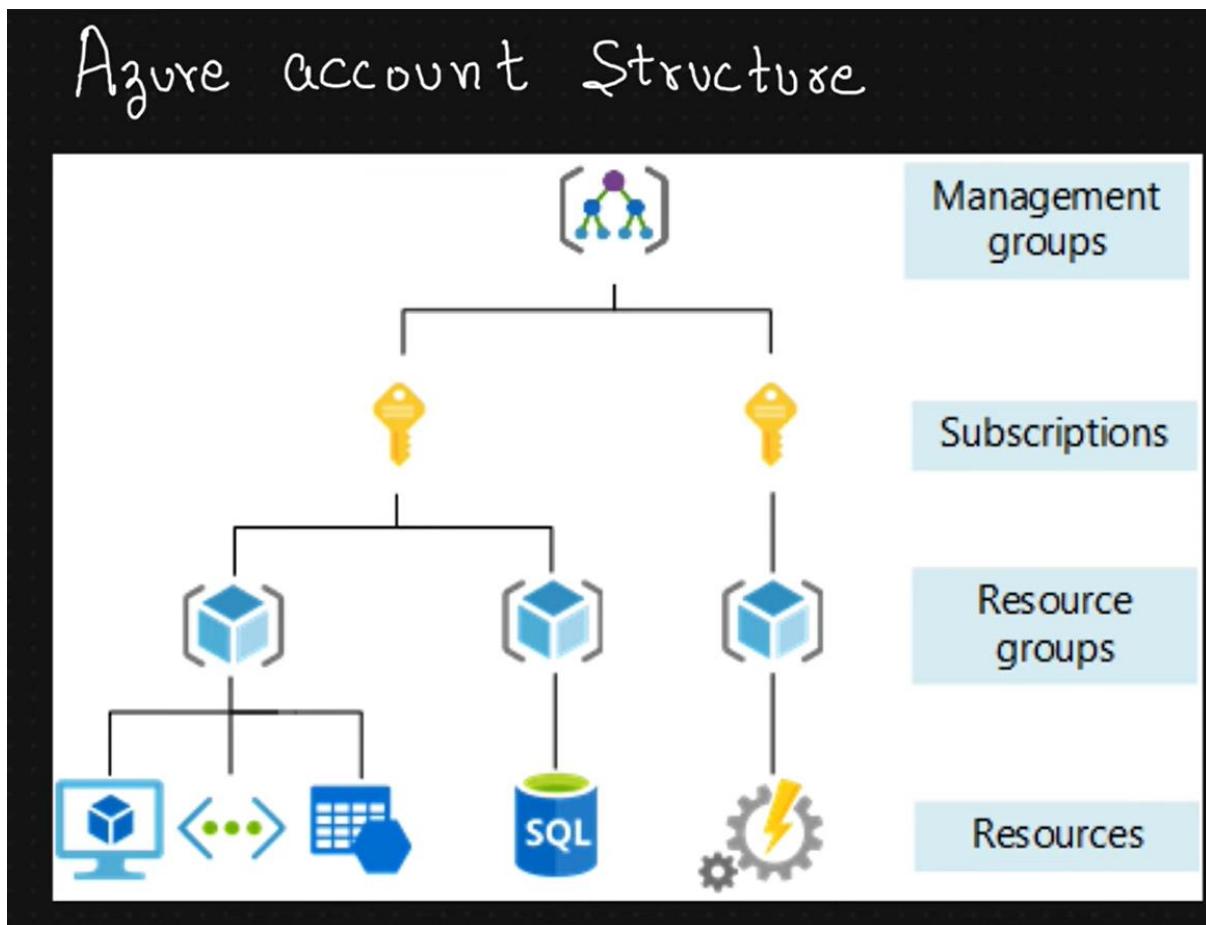


Domain – Ecommerce Data Engineering



Azure Account Structure



→ it already free train in student id

Screenshot of the Microsoft Azure portal showing the Azure for Students subscription details:

Subscription Overview:

- Subscription Name:** Azure for Students
- Subscription ID:** d7c1e8a1-8bd2-4126-ba88-75140d084a4d
- Directory:** Default Directory (dineshkumar7283@gmail.onmicrosoft.com)
- Role:** Account admin
- Offer:** Azure for Students
- Offer ID:** MS-AZR-0170P
- Parent management group:** 1eadda0c-f714-4f5e-8af6-f435b63d1d8e
- Currency:** INR
- Status:** Active
- Secure Score:** Not available

Subscription Filter:

- Subscriptions : Filtered (1 of 1)
- My role == all
- Status == all
- Add filter

Subscription Name: Azure for Students

Top products by number of resources:

Product	Count
Product A	5
Product B	4
Product C	2
Product D	1
Product E	1

Azure Defender coverage:

A chart showing Azure Defender coverage across various categories.

→Create Azure Resource Group

The screenshot shows the Microsoft Azure portal interface for creating a new resource group. The URL in the address bar is portal.azure.com/#create/Microsoft.ResourceGroup. The page title is "Create a resource group". There are three tabs at the top: "Basics", "Tags", and "Review + create". The "Basics" tab is selected. A note below the tabs explains what a resource group is: "A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization." Below this note are three input fields: "Subscription" (set to "Azure for Students"), "Resource group name" (set to "Ecomm-live"), and "Region" (set to "(Asia Pacific) Central India"). At the bottom of the form are three buttons: "Previous", "Next", and "Review + create".

The screenshot shows the Microsoft Azure portal interface for the "Ecomm-live" resource group. The URL in the address bar is portal.azure.com/@#dineshdkumar7283@gmail.onmicrosoft.com/resource/subscriptions/d7c1e8a1-8bd2-4126-ba88-75140d084a4d/resourceGroups/Ecomm-live/overview. The page title is "Ecomm-live". The left sidebar shows navigation options like "Overview", "Activity log", "Access control (IAM)", "Tags", "Resource visualizer", "Events", "Settings", "Cost Management", "Monitoring", "Automation", and "Help". The main content area is titled "Overview" and includes sections for "Essentials" (Subscription: Azure for Students, Deployment: No deployments, Location: Central India), "Resources" (with filters applied: Type equals all, Location equals all), and "Recommendations". A message at the bottom states "No resources match your filters".

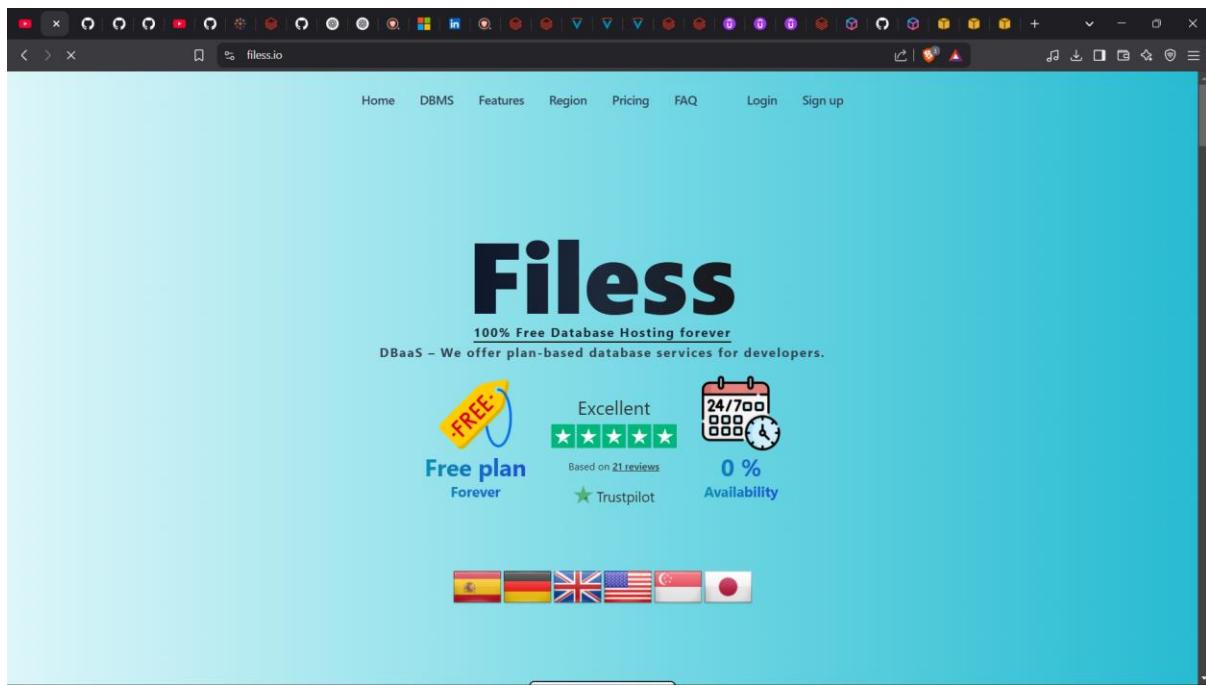
→Dataset used

The screenshot shows the Kaggle website interface. On the left, there's a sidebar with navigation links: Create, Home, Competitions, Datasets (which is selected), Models, Code, Discussions, Learn, and More. The main content area displays a dataset titled "Brazilian E-Commerce Public Dataset by Olist". It features a search bar at the top right, a sign-in/register button, and a download button. Below the title, it says "OLIST AND 3 COLLABORATORS - UPDATED 3 YEARS AGO" and "3440". A "Data Card" section includes links for Data Card, Code (570), Discussion (60), and Suggestions (0). To the right, there's a "About Dataset" section with a detailed description of the dataset, mentioning 100,000 orders from 2016 to 2018 across multiple marketplaces in Brazil. It also includes sections for Usability (10.00), License (CC BY-NC-SA 4.0), Expected update frequency (Never), and Tags (Business, Data Visualization). At the bottom, a message about cookie usage is shown, along with "Learn more" and "OK, Got it" buttons.

The screenshot shows a GitHub repository page for "dinesh6351 / Databricks_Ecomm_Krish_Capstone". The repository has a main branch named "main". The "Data" folder is expanded, showing several CSV files: olist_customers_dataset.csv, olist_geolocation_dataset.csv, olist_order_items_dataset.csv, olist_order_payments_dataset.csv, olist_order_reviews_dataset.csv, olist_orders_dataset.csv, olist_products_dataset.csv, olist_sellers_dataset.csv, product_category_name_transl..., and DS_Store. A commit history for "dinesh7283 first commit" is visible, showing the creation of these files. The commit was made 48fa713 - 3 minutes ago. The GitHub interface includes a search bar, a file browser, and various repository management buttons like Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings.

→use one dataset in SQL as well

→use this website



→ Register and login

A screenshot of a web browser showing the dashboard at dash.filess.io/#/app/databases. The dashboard has a light blue header with the filess.io logo and a user profile. The main area is titled "Databases". On the left, there's a sidebar with "Databases" selected, followed by "Standard Tier", "Billing", "API", and "Status". A "Get more?" section with an "Upgrade Now" button is also present. The main content area shows a table with columns: Motor, Identifier, Available, and Location. A search bar at the top of the table says "Search DB name...". Below the table, it says "This table is empty". At the bottom, there are pagination controls: "Rows per page: 5", "0-0 of 0", and arrows. A blue circular icon is in the bottom right corner.

→ Databases

- 1) SQL
- 2) NoSQL
- 3) HTTPS

The screenshot shows the filess.io dashboard with the URL `dash.filess.io/#/app/databases/new/?mode=standard`. On the left, there's a sidebar with a user profile (dinesh km), navigation links for Databases, Standard Tier, Billing, API, and Status, and a 'Get more?' section with an 'Upgrade Now' button. The main area is titled 'Create a new database'. It has two tabs: 'Standard Database' (selected) and 'Dedicated Database'. The 'Database identifier' field contains 'olistproject'. A note says 'You can't choose a region because you are on the Free tier. The region of your database will be chosen automatically from the list below. If you want to choose a region, and see full list of regions, please upgrade your tier.' Below this, the 'Database engine' is set to 'MySQL v8.0.29' with a 'Clear' button. A list of regions is shown: Spain (sp-2), Düsseldorf (dus-1), Nürnberg (nu-1), and Nürnberg (nu-2). A large blue circular button is at the bottom right.

→ Click create

→ Create same as mongo DB

This screenshot is similar to the previous one but with different database settings. The 'Database identifier' field now contains 'olistDataNoSQL'. The 'Database engine' is set to 'Mongo v7.0.2' with a 'Clear' button. The list of regions includes additional entries: Spain (sp-2), Düsseldorf (dus-1), Nürnberg (nu-1), Nürnberg (nu-2), Portsmouth (uk-1), and Portsmouth (uk-2). A large blue circular button is at the bottom right.

The screenshot shows the filess.io dashboard interface. On the left, there's a sidebar with navigation links: 'Databases' (selected), 'Standard Tier', 'Billing', 'API', and 'Status'. A 'Get more?' button with a 'Upgrade Now' link is also present. The main area is titled 'Databases' and shows two entries in a table:

	Motor	Identifier	Available	Location
1	MuDB v8.0.29	olistproject	Yes	Germany
2	Redis v7.0.2	olistDataNoSQL	Yes	Germany

At the bottom right of the main area, there are buttons for 'Rows per page:' (set to 5), '1-2 of 2', and navigation arrows. The top right of the dashboard includes a user profile icon, a notification badge (1), and a 'New Database' button.

→ use googlecolab to read data and write in SQL

The screenshot shows a Google Colab notebook titled 'DataIntentionToSQL.ipynb'. The code cell contains the following Python code:

```
[1]: print("HI")
```

The output of the cell is:

```
HI
```

The notebook interface includes a file browser on the left showing files like 'sample_data' and 'olist_order_payments_dataset...', and various status indicators at the bottom like 'Disk 70.92 GB available' and a completion message '0s completed at 5:02PM'.

filess.io

dinesh km
dineshkumar63519
500@gmail.com
Free tier

Databases

Standard Tier

Billing

API

Status

euyak.h.filess.io

Database: olistproject_tightdaily User: olistproject_tightdaily Port: 3307

Do not drop this database. It won't be included in the backup if it does not exist.

Password:

MySQL URI: mysql://olistproject_tightdaily:...@euyak.h.filess.io:3307/olistproject_tightdaily

MySQL login command: mysql -u olistproject_tightdaily -P 3307 -p... -h euyak.h.filess.io olistproject_tightdaily

Connection formats

Request Backup

Delete Olistproject

Backups

→ click code and change python

filess.io

dinesh km
dineshkumar63519
500@gmail.com
Free tier

Databases

Standard Tier

Billing

API

Status

Get more storage and more features.

Generate code

Generate code to connect to your database. You can use this code in your application to connect to your database.

Language: Python

```
1 import mysql.connector
2 from mysql.connector import Error
3
4 hostname = "euyak.h.filess.io"
5 database = "olistproject_tightdaily"
6 port = "3307"
7 username = "olistproject_tightdaily"
8 password = "c6756596e0c42d4f639647ea7de2458e61d8306"
9
10 try:
11     connection = mysql.connector.connect(host=hostname, database=database, user=username, password=password, port=port)
12     if connection.is_connected():
13         db_Info = connection.get_server_info()
14         print("Connected to MySQL Server version ", db_Info)
15         cursor = connection.cursor()
16         cursor.execute("select database();")
17         record = cursor.fetchone()
18         print("You're connected to database: ", record)
19
20     except Error as e:
21         print("Error while connecting to MySQL", e)
22     finally:
23         if connection.is_connected():
```

Copy Code

→ Paste in colab

The screenshot shows a Google Colab interface with a notebook titled "DataIntentionToSQL.ipynb". The code cell contains a Python script for connecting to a MySQL database:

```
import mysql.connector
from mysql.connector import Error

hostname = "euyak.h.fileless.io"
database = "olistproject_tightdaily"
port = "3307"
username = "olistproject_tightdaily"
password = "c67565996e0cA2d4f639647ea7de2458e61d8306"

try:
    connection = mysql.connector.connect(host=hostname, database=database, user=username, password=password, port=port)
    if connection.is_connected():
        db_info = connection.get_server_info()
        print("Connected to MySQL Server version ", db_info)
        cursor = connection.cursor()
        cursor.execute("select database();")
        record = cursor.fetchone()
        print("You're connected to database: ", record)

except Error as e:
    print("Error while connecting to MySQL", e)
finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed")
```

The screenshot shows the same Google Colab interface after running the command `!pip install mysql-connector-python`. The output indicates the package is being downloaded and installed successfully:

```
[2] !pip install mysql-connector-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-9.2.0-cp311-cp311-manylinux_2_28_x86_64.whl.metadata (6.0 kB)
  Downloading mysql_connector_python-9.2.0-cp311-cp311-manylinux_2_28_x86_64.whl (34.0 kB)
    34.0/34.0 MB 35.0 MB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.2.0
```

Below the output, the same Python script is shown again.

```
import pandas as pd
import mysql.connector
from mysql.connector import Error

hostname = "euyak.h.fileless.io"
database = "olistproject_lightdaily"
port = 3307
username = "olistproject_lightdaily"
password = "c67565996e0c42d4f639647ea7de2458e61d8306"

# CSV file path
csv_file_path = "olist_order_payments_dataset.csv"

# Table name where the data will be uploaded
table_name = "olist_order_payments"

try:
    # Step 1: Establish a connection to MySQL server
    connection = mysql.connector.connect(
        host=hostname,
        database=database,
        user=username,
        password=password,
        port=port
    )
    if connection.is_connected():
        print("Connected to MySQL Server successfully!")

        # Step 2: Create a cursor to execute SQL queries
        cursor = connection.cursor()

        # Step 3: Drop table if it already exists (for clean insertion)
        cursor.execute(f"DROP TABLE IF EXISTS {table_name};")
        print(f"Table '{table_name}' dropped if it existed.")

        # Step 4: Create a table structure to match CSV file
        create_table_query = f"""
CREATE TABLE {table_name} (
    order_id VARCHAR(50),
    payment_sequential INT,
    payment_type VARCHAR(20),
    payment_installments INT,
    payment_value FLOAT
);
"""
        cursor.execute(create_table_query)
        print(f"Table '{table_name}' created successfully!")

        # Step 5: Load the CSV data into pandas DataFrame
        data = pd.read_csv(csv_file_path)
        print("CSV data loaded into pandas DataFrame.")

        # Step 6: Insert data in batches of 500 records
        batch_size = 500 # Define the batch size
        total_records = len(data) # Get total records in the DataFrame
        print(f"Starting data insertion into '{table_name}' in batches of {batch_size} records.")
        for start in range(0, total_records, batch_size):
            end = start + batch_size
            batch = data.iloc[start:end] # Get the current batch of records

            # Convert batch to list of tuples for MySQL insertion
            batch_records = [
                tuple(row) for row in batch.itertuples(index=False, name=None)
            ]

```

```
cursor.execute(batch_records)
print(f"{len(batch_records)} rows inserted into {table_name} successfully!")

finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection closed successfully!")

```

```
# Prepare the INSERT query
insert_query = f"""
INSERT INTO {table_name}
(order_id, payment_sequential, payment_type, payment_installments, payment_value)
VALUES (%s, %s, %s, %s, %s);
"""

# Execute the insertion query for the batch
cursor.executemany(insert_query, batch_records)
connection.commit() # Commit after each batch
print(f"Inserted records {start + 1} to {min(end, total_records)} successfully.")

print(f"All {total_records} records inserted successfully into '{table_name}'.")

except Error as e:
    # Step 7: Handle any errors
    print("Error while connecting to MySQL or inserting data:", e)

finally:
    # Step 8: Close the cursor and connection
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed.")
```

```
Connected to MySQL Server successfully!
Table 'olist_order_payments' dropped if it existed.
Table 'olist_order_payments' created successfully!
CSV data loaded into pandas DataFrame.
Starting data insertion into 'olist_order_payments' in batches of 500 records.
Inserted records 1 to 500 successfully.
Inserted records 501 to 1000 successfully.
Inserted records 1001 to 1500 successfully.
Inserted records 1501 to 2000 successfully.
Inserted records 2001 to 2500 successfully.
Inserted records 2501 to 3000 successfully.
Inserted records 3001 to 3500 successfully.
Inserted records 3501 to 4000 successfully.
Inserted records 4001 to 4500 successfully.
Inserted records 4501 to 5000 successfully.
Inserted records 5001 to 5500 successfully.
Inserted records 5501 to 6000 successfully.
Inserted records 6001 to 6500 successfully.
Inserted records 6501 to 7000 successfully.
Inserted records 7001 to 7500 successfully.
```

CONNECTIONS

Search connection or database

olistproject_tightdaily on euyak.h.filess

Add new connection

TABLES, VIEWS, FUNCTIONS

Search in tables, objects, # prefix in column

Tables (1)

olist_order_payments 1,000 rows

order_id varchar(50)
payment_sequential int
payment_type varchar(20)
payment_installments int
payment_value float

client.filess.io

CONNECTIONS

Search connection or database

olistproject_tightdaily on euyak.h.filess

Add new connection

TABLES, VIEWS, FUNCTIONS

Search in tables, objects, # prefix in column

Tables (1)

olist_order_payments 1,000 rows

order_id varchar(50)
payment_sequential int
payment_type varchar(20)
payment_installments int
payment_value float

Messages Result 1

COLUMNS

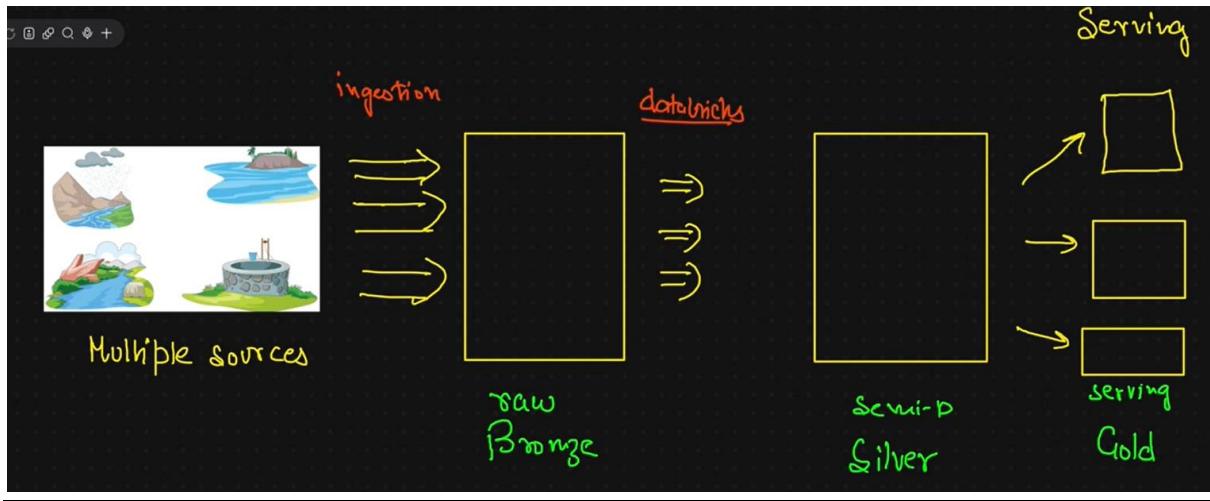
order_id payment_sequential payment_type payment_installments

	order_id	payment_sequential	payment_type	payment_installments
1	b81ef226f3fe1789b1e8b2acac839d17	1	credit_card	8
2	a9810da62917af2d9aef1d278ff1dcfa0	1	credit_card	1
3	25e0ea4e93396b6fa0d3dd708e76c1bd	1	credit_card	1
4	ba78997921bbcd1373bb41e913ab953	1	credit_card	8
5	42fdff800a16b4769251d4d99d4441a	1	credit_card	2
6	298fcdf1f73eb413ae4d26d01b25bc1cd	1	credit_card	2
7	771ee386b001f06208a7419e4fc1bbd	1	credit_card	1
8	3d7239c394a212faae122962d5f14ac7	1	credit_card	3
9	1f78449c7a54fa9e96e88ba1491fa9	1	credit_card	6
10	0572b5c522d4708006550a45b4-6714	1	poloto	1

Execute Kill Save Format code Export result

client.filess.io

Create pipeline form different lake to store some where



Go to Azure

The screenshot shows the Microsoft Azure portal interface for the 'Ecomm-live' resource group. The top navigation bar includes links for Home, Resource groups, and the current group. The main content area has a search bar and several management buttons. On the left, a sidebar lists options like Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, Settings, Cost Management, Monitoring, Automation, and Help. The central area displays resource details such as Subscription (move: Azure for Students), Subscription ID, Location, and Tags. Below this, a 'Resources' section shows a table with columns for Name, Type, and Location, with a note that 0 records were found. A large message in the center states 'No resources match your filters' with a link to 'Try changing or clearing your filters.' At the bottom, there are buttons for '+ Create resources' and 'Clear filters'.

→create datafactory

The screenshot shows the Microsoft Azure Marketplace page for the 'Data Factory' service. At the top, there's a navigation bar with links for Home, Resource groups, Ecomm-live, Marketplace, and Data Factory. Below the navigation is a search bar and a Copilot button. The main content area features a large thumbnail for the Data Factory service, which includes a 'Data Factory' icon, the service name, a 'Add to Favorites' link, a Microsoft Azure Service badge, a rating of 3.6 (607 ratings), and a 'Create' button. Below the thumbnail are tabs for Overview, Plans, Usage Information + Support, and Ratings + Reviews. The Overview tab is selected. The content under Overview describes the service as a serverless integration service for data integration needs, mentioning ETL and ELT processes, native connectors, and SSIS integration runtime. It also lists several benefits: no code or maintenance required, cost-efficient, fully managed, and scales on demand. A screenshot of the Azure Data Factory visual environment is shown, displaying a pipeline with various stages and data flows. At the bottom right of the screenshot is a 'Give feedback' button.

The screenshot shows the 'Create Data Factory' wizard in the Microsoft Azure portal. The top navigation bar includes Home, Resource groups, Ecomm-live, Marketplace, Data Factory, Create Data Factory, and a '...' button. The main content area has a header 'Create Data Factory' with a '...' button. Below the header are tabs for Basics, Git configuration, Networking, Advanced, Tags, and Review + create. The Basics tab is selected. The 'Project details' section asks to select a subscription and resource group. The 'Subscription' dropdown is set to 'Azure for Students' and the 'Resource group' dropdown is set to 'Ecomm-live' with a 'Create new' option. The 'Instance details' section requires entering a Name (set to 'olist-dinesh-data-factory'), Region (set to 'Central India'), and Version (set to 'V2'). At the bottom of the form are 'Previous' and 'Next' buttons, and a 'Review + create' button. A 'Give feedback' button is located at the bottom right.

TERMS

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

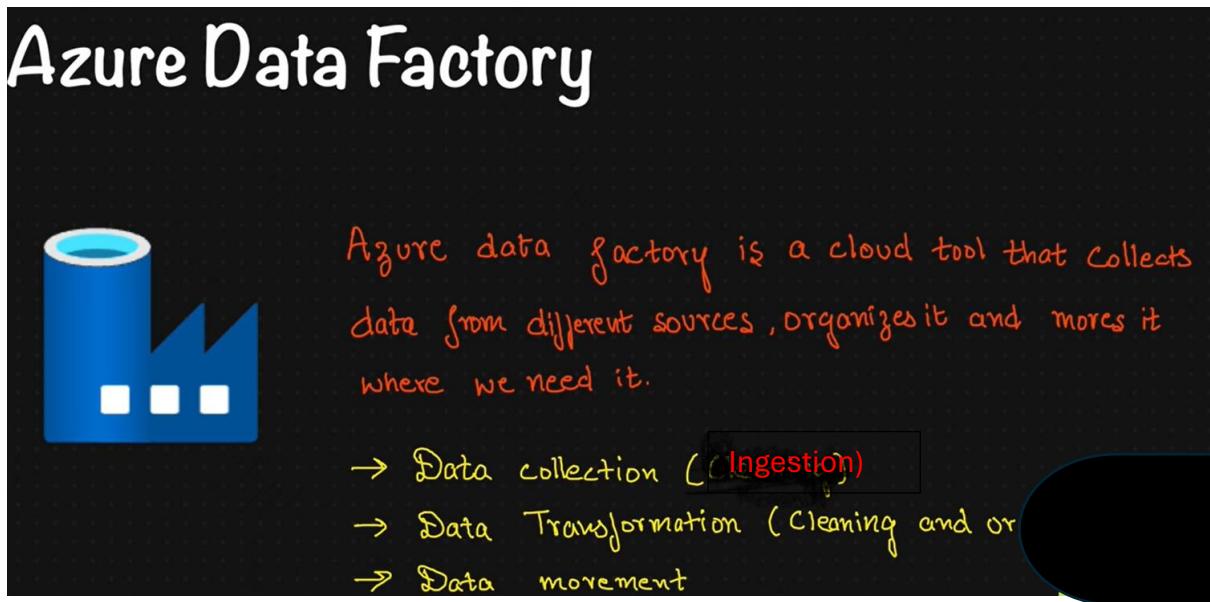
Basics

Subscription	Azure for Students
Resource group	Ecomm-live
Name	olist-ecomm-data-factory-dinesh
Region	Central India
Version	V2

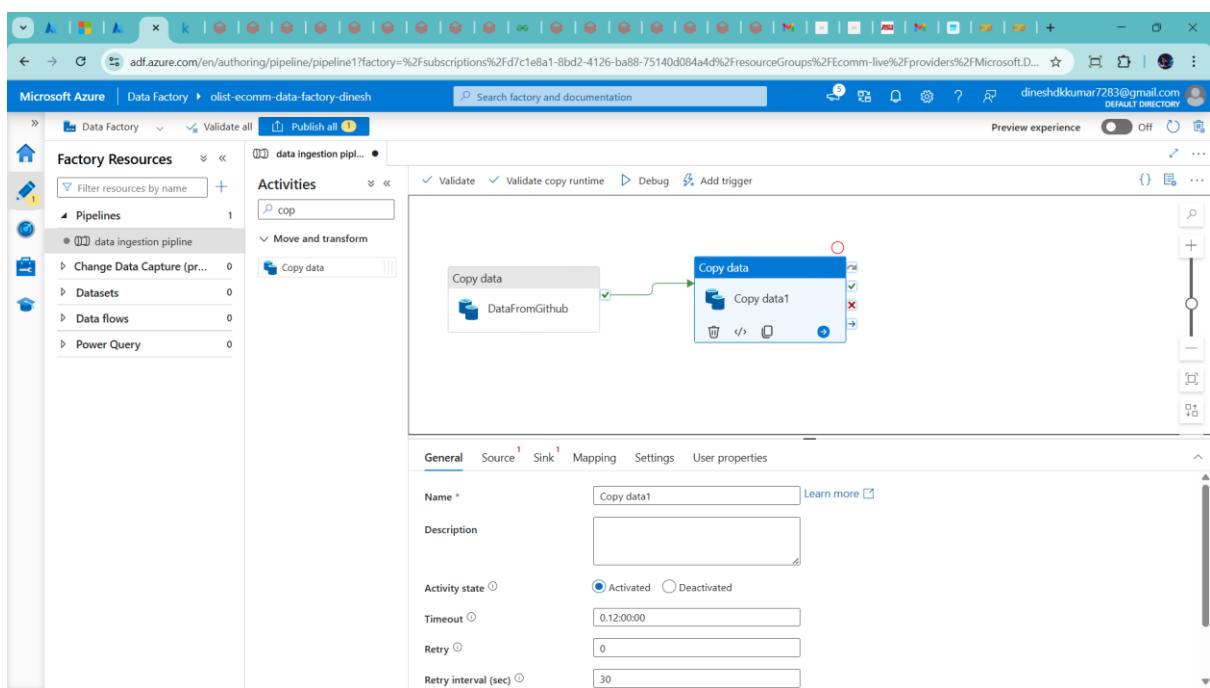
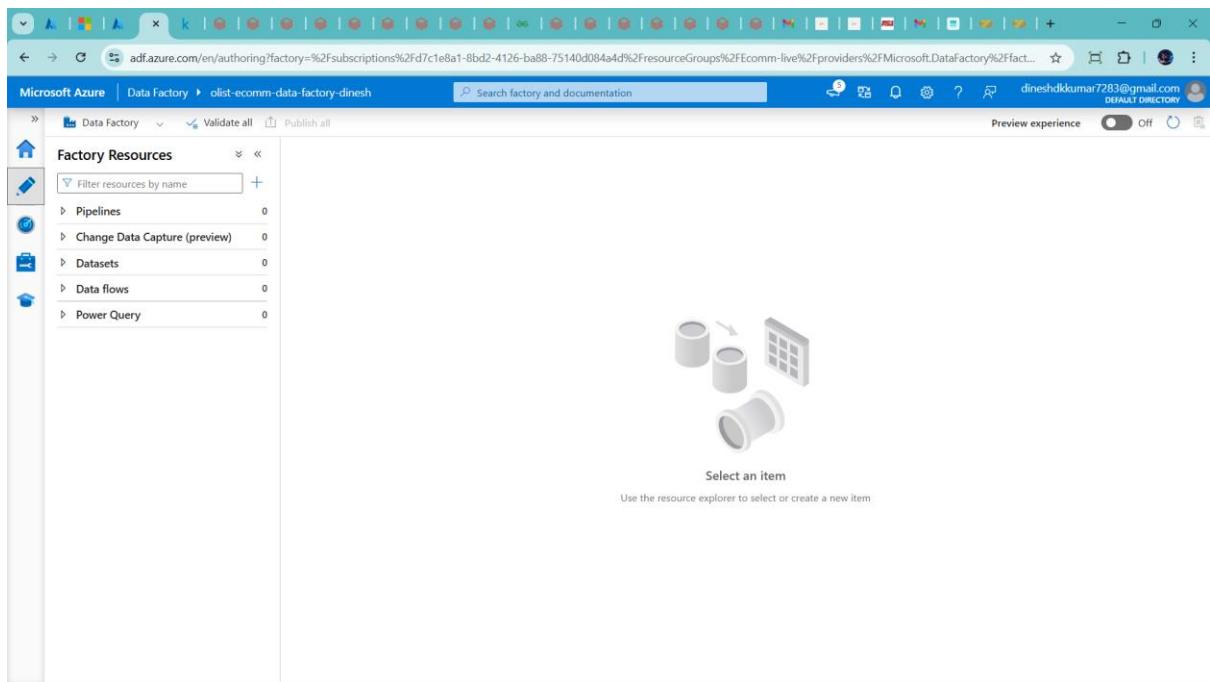
Networking

Connect via	Public endpoint
-------------	-----------------

[Previous](#) [Next](#) [Create](#) [Give feedback](#)

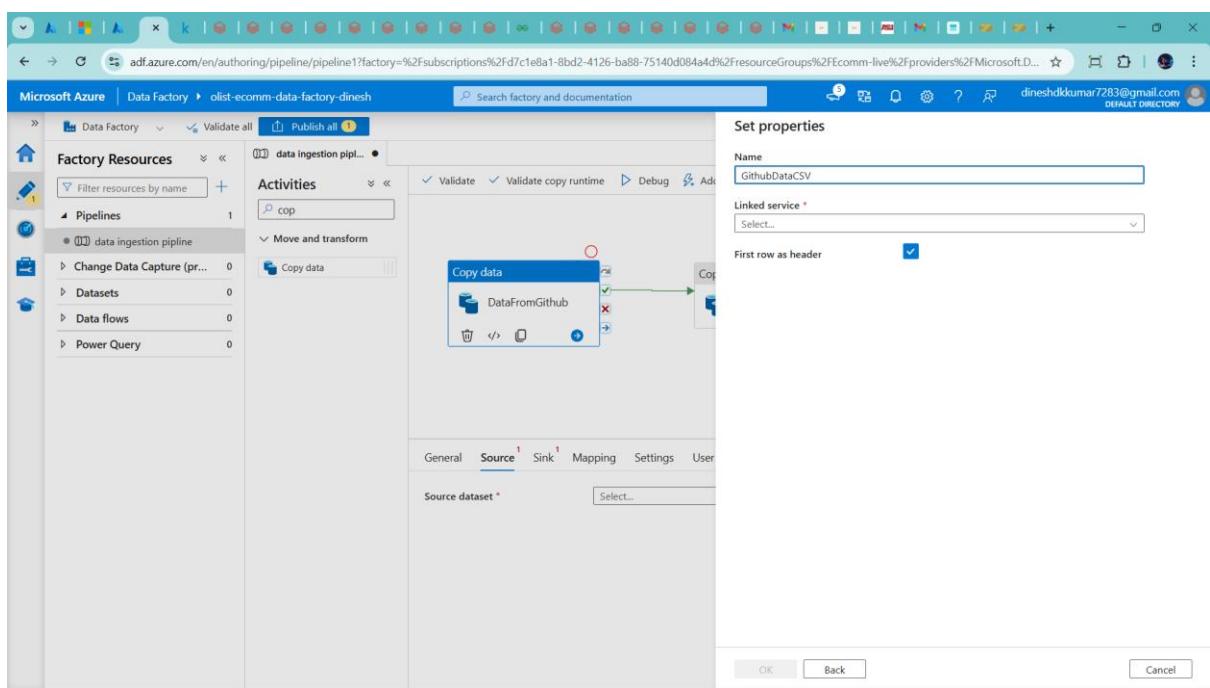
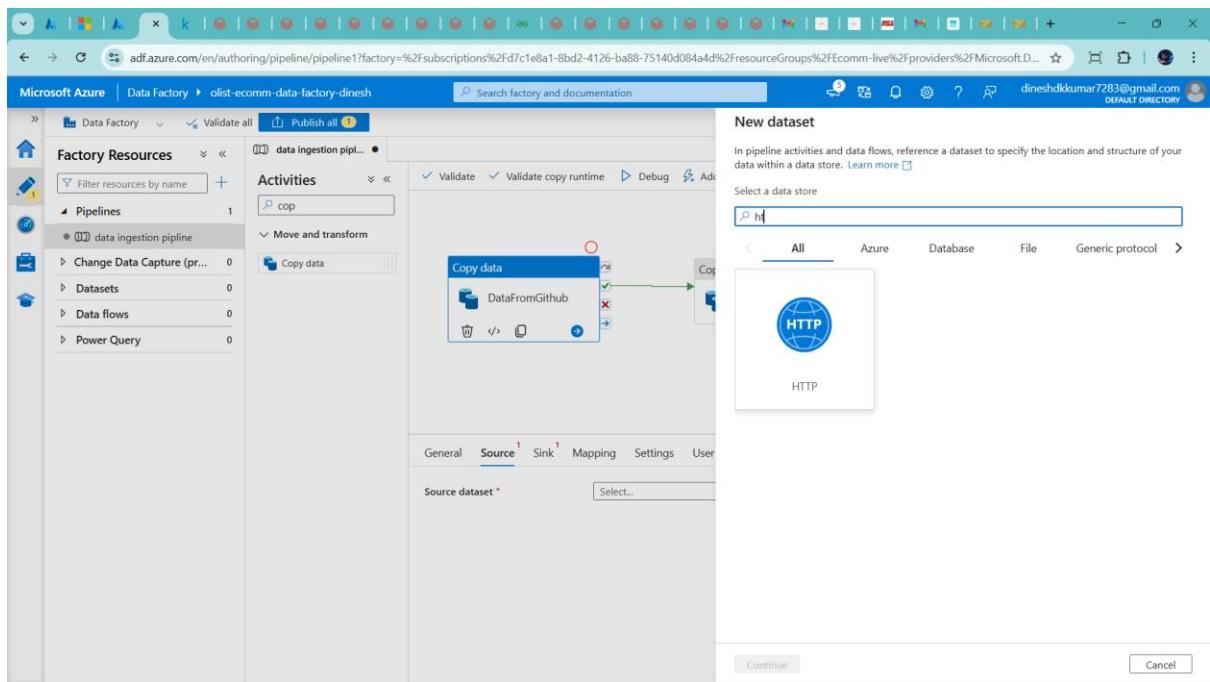


-→Launch Data Factory

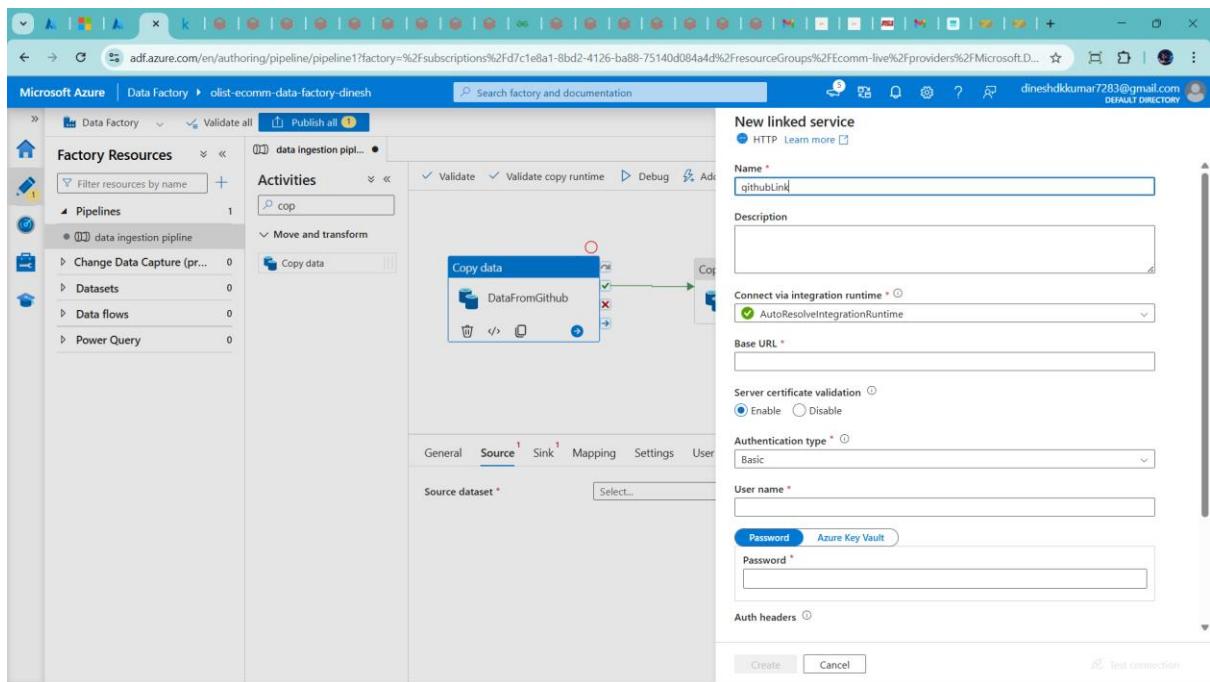


→ Click Source → click + new

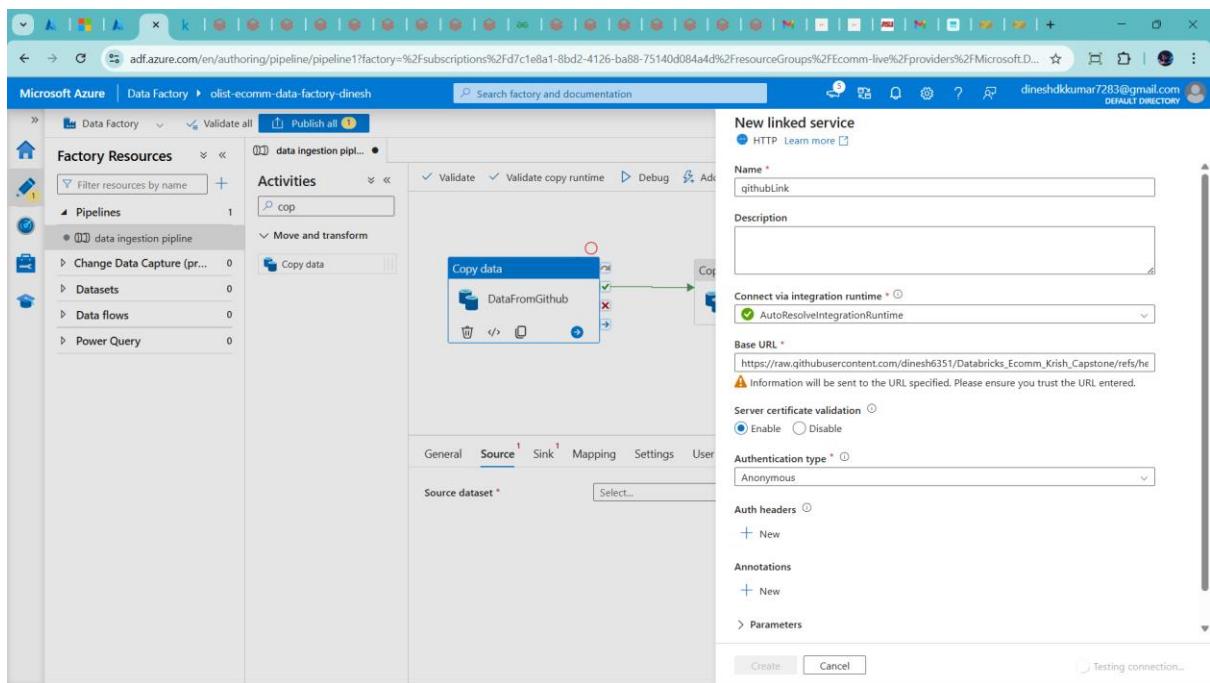
→ choose https → csv

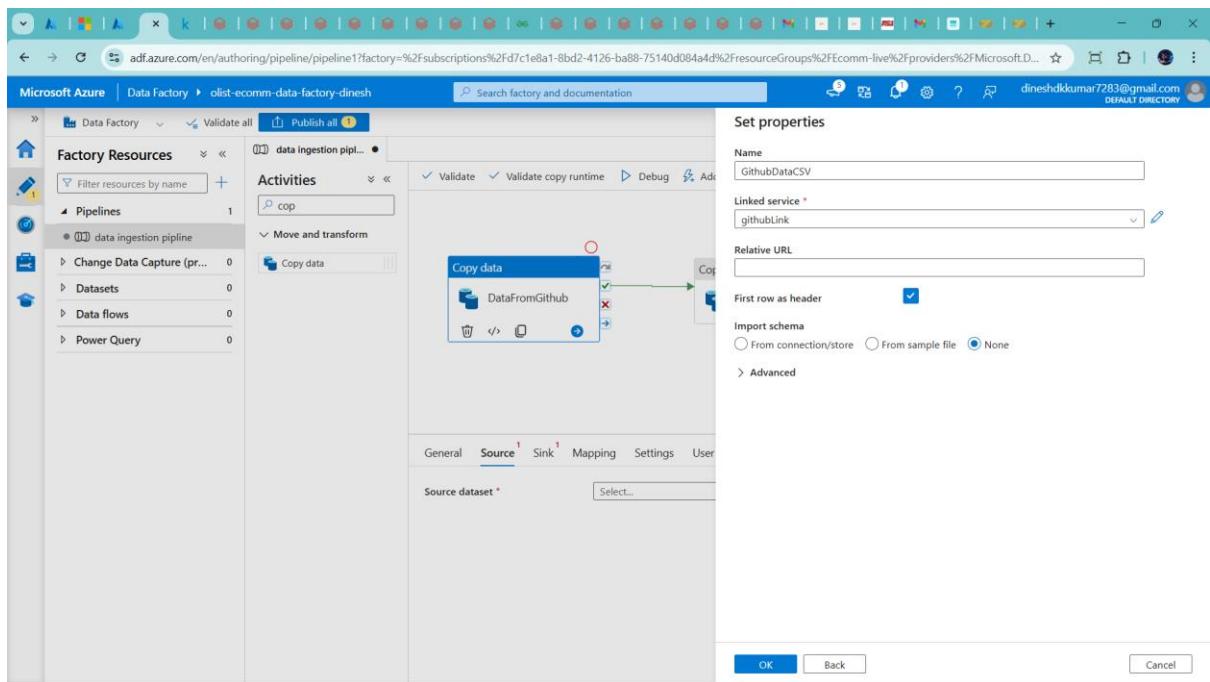


→click linked service -> +New

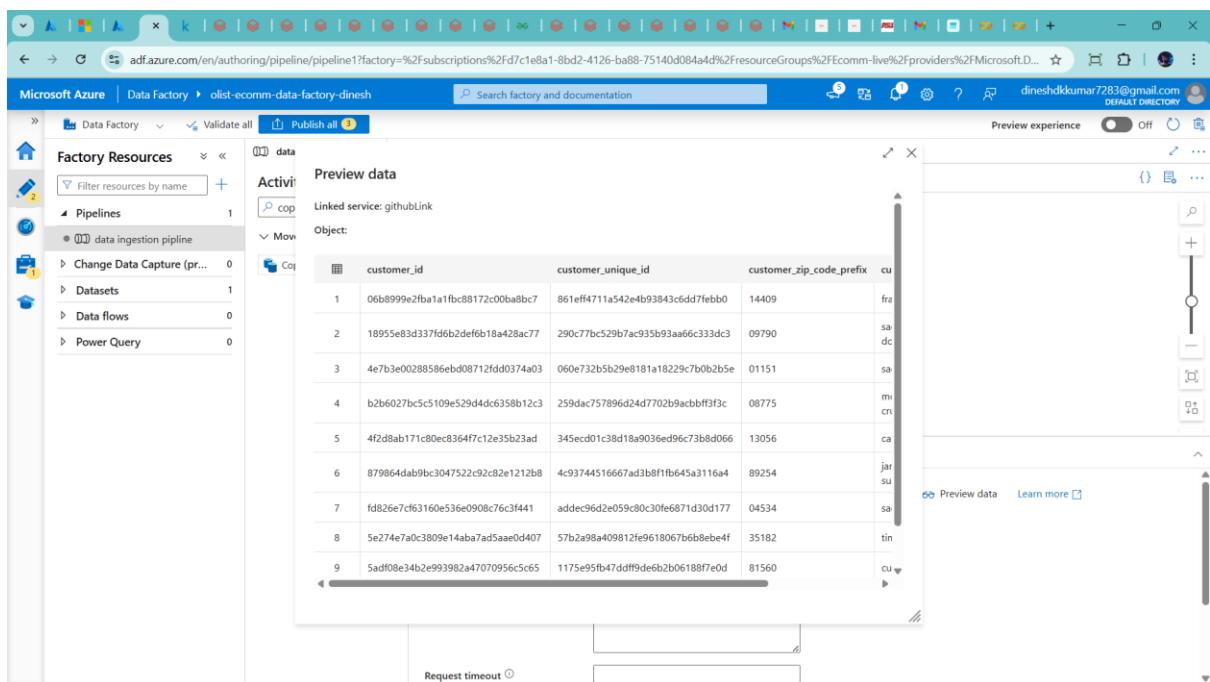


→ in this base url add full url link





→ Click ok



→ Change its Base URL

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar is visible with 'Pipelines', 'Datasets', and 'Power Query' sections. In the center, a 'data ingestion pipeline' is selected. A 'GithubDataCSV' dataset is highlighted. On the right, the 'Edit linked service' dialog for 'githubLink' is open. The 'Base URL' field contains the URL 'https://raw.githubusercontent.com/dineshkumar7283/ecommerce/1.0/o...'. Other settings like 'Compression type', 'Column delimiter', and 'Row delimiter' are also visible.

The screenshot shows the Microsoft Azure Data Factory interface. The 'Factory Resources' sidebar is visible. In the center, a 'data ingestion pipeline' is selected. A 'GithubDataCSV' dataset is highlighted. On the right, the 'Properties' panel is open for 'GithubDataCSV'. The 'General' tab shows the 'Name' as 'GithubDataCSV'. Other tabs like 'Related (1)' and 'Annotations' are also visible.

→ where you store destination storage → Sink

→ Now create storage account

The screenshot shows the Microsoft Azure Marketplace page for the Storage account service. At the top, there's a navigation bar with links for Home, Resource groups, Marketplace, and Storage account. Below the navigation is a search bar and a Copilot button. The main content area has a title 'Storage account' with a 'Create' button. A 'Plan' dropdown is set to 'Storage account'. Below this, there are tabs for Overview, Plans, Usage Information + Support, and Ratings + Reviews. The Overview tab is selected. It contains a brief description of Microsoft Azure Storage, mentioning scalability, durability, and recovery solutions for data. There are also sections for 'More products from Microsoft' featuring Active Directory Health Check, AD Replication Status, Device Update for IoT Hub, and Front Door and CDN profiles.

The screenshot shows the 'Create a storage account' wizard on the Microsoft Azure portal. The URL in the address bar is <https://portal.azure.com/#create/Microsoft.StorageAccount-ARM>. The page header includes the Microsoft Azure logo, a search bar, and a Copilot button. The main content area is titled 'Create a storage account'. It starts with a note about the cost of storage accounts. The 'Project details' section requires selecting a subscription ('Azure for Students') and a resource group ('Ecomm-live'). The 'Instance details' section includes fields for 'Storage account name' (set to 'olidatastoragedinesh'), 'Region' (set to '(Asia Pacific) Central India'), 'Primary service' (dropdown menu), 'Performance' (radio buttons for 'Standard' and 'Premium', with 'Standard' selected), and 'Redundancy' (dropdown menu set to 'Locally-redundant storage (LRS)'). At the bottom, there are 'Previous' and 'Next' buttons, and a 'Review + create' button.

→ Redundancy → how duplicate copy store different location

→ Enable Hierarchical → it possible to create folder

Create a storage account

Require secure transfer for REST API operations

Allow enabling anonymous access on individual containers

Enable storage account key access

Default to Microsoft Entra authorization in the Azure portal

Minimum TLS version Version 1.2

Permitted scope for copy operations (preview) From any 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

[Previous](#) [Next](#) [Review + create](#) [Give feedback](#)

→ Click review +create

Create a storage account

[View automation template](#)

Basics

Subscription	Azure for Students
Resource group	Ecomm-live
Location	Central India
Storage account name	olistdatastoragedinesh
Primary service	
Performance	Standard
Replication	Locally-redundant storage (LRS)

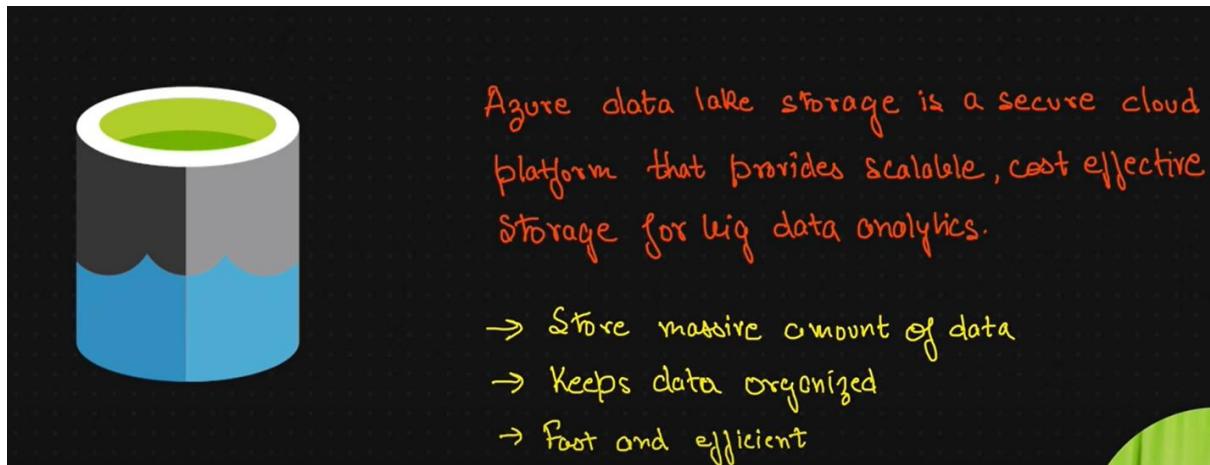
Advanced

Enable hierarchical namespace	Enabled
Enable SFTP	Disabled
Enable network file system v3	Disabled
Allow cross-tenant replication	Disabled
Access tier	Hot
Enable large file shares	Enabled

Security

[Previous](#) [Next](#) [Create](#) [Give feedback](#)

→ Create



Screenshot of the Microsoft Azure portal showing the 'Ecomm-live' resource group overview. The portal interface includes a navigation bar, search bar, and various management tools. The main content area displays subscription details (move to 'Azure for Students'), deployment status (2 succeeded), and location (Central India). A 'Resources' section lists two items: 'olist-ecomm-data-factory-dinesh' (Data factory V2) and 'olistdatastoragedinesh' (Storage account), both located in Central India.

Name	Type	Location	Actions
olist-ecomm-data-factory-dinesh	Data factory (V2)	Central India	...
olistdatastoragedinesh	Storage account	Central India	...

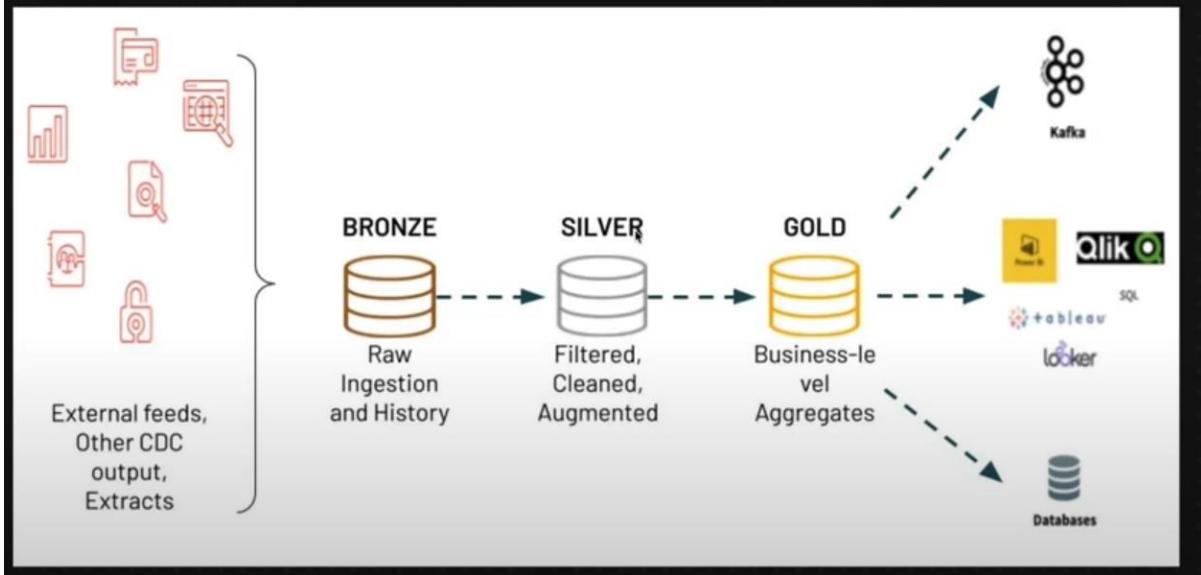
→ create container

The screenshot shows the Microsoft Azure Storage Container creation interface. On the left, there's a sidebar with various storage-related options like Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage browser, Partner solutions, Resource visualizer, and Data storage (Containers, File shares, Queues, Tables). The 'Containers' option under Data storage is selected. The main area shows a table with one row: Name (Slogs), Last modified (3/18/2025, 3:00:40 PM), and Anonymous (Private). A search bar at the top says 'Search containers by prefix'. To the right, a 'New container' dialog is open with a 'Name' field containing 'olistdata' and an 'Anonymous access level' dropdown set to 'Private (no anonymous access)'. A note below says 'The access level is set to private because anonymous access is disabled on this storage account.' At the bottom right of the dialog are 'Create' and 'Give feedback' buttons.

→add directory (bronze,silver,gold)

The screenshot shows the Microsoft Azure Storage Container overview page for the 'olistdata' container. The sidebar on the left is similar to the previous screen. The main area shows a table of blobs with three entries: 'bronze' (Modified 3/18/2025, 3:04:55 PM), 'gold' (Modified 3/18/2025, 3:05:09 PM), and 'silver' (Modified 3/18/2025, 3:05:02 PM). Above the table, it says 'Successfully added directory' and 'Successfully added directory 'gold''. The table has columns: Name, Modified, Access tier, Archive status, Blob type, Size, and Lease state. There are also buttons for Upload, Add Directory, Refresh, Rename, Delete, Change tier, Acquire lease, Break lease, and Give feedback.

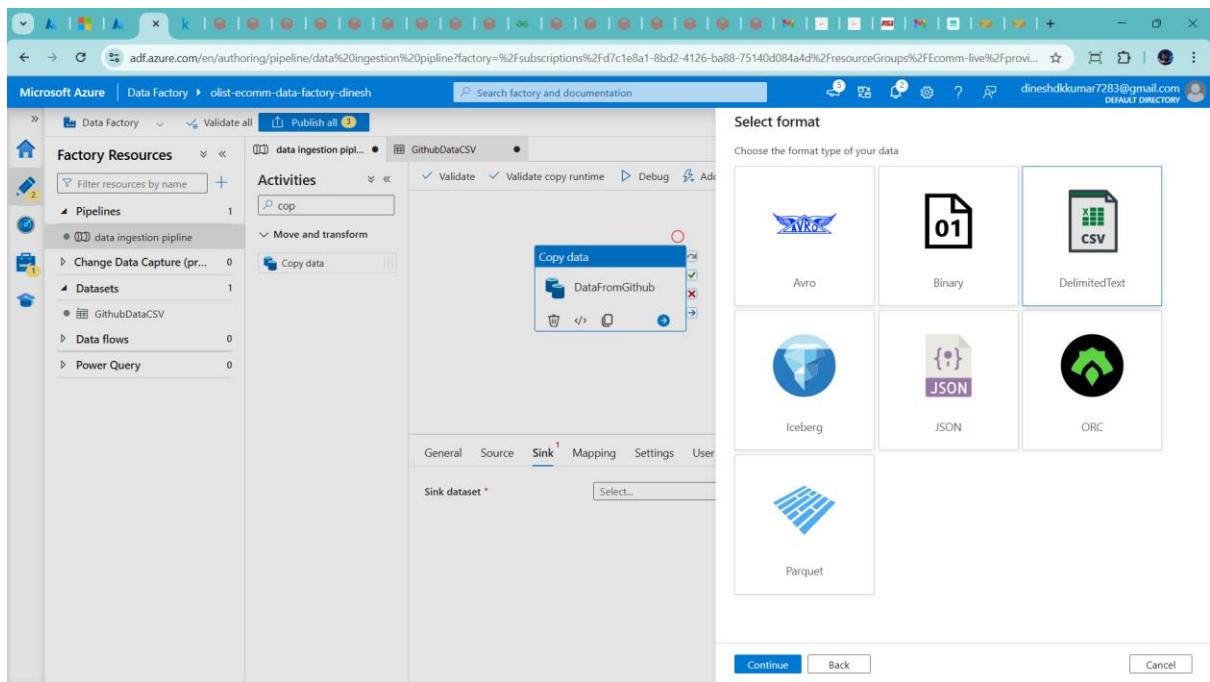
Medallion Architecture



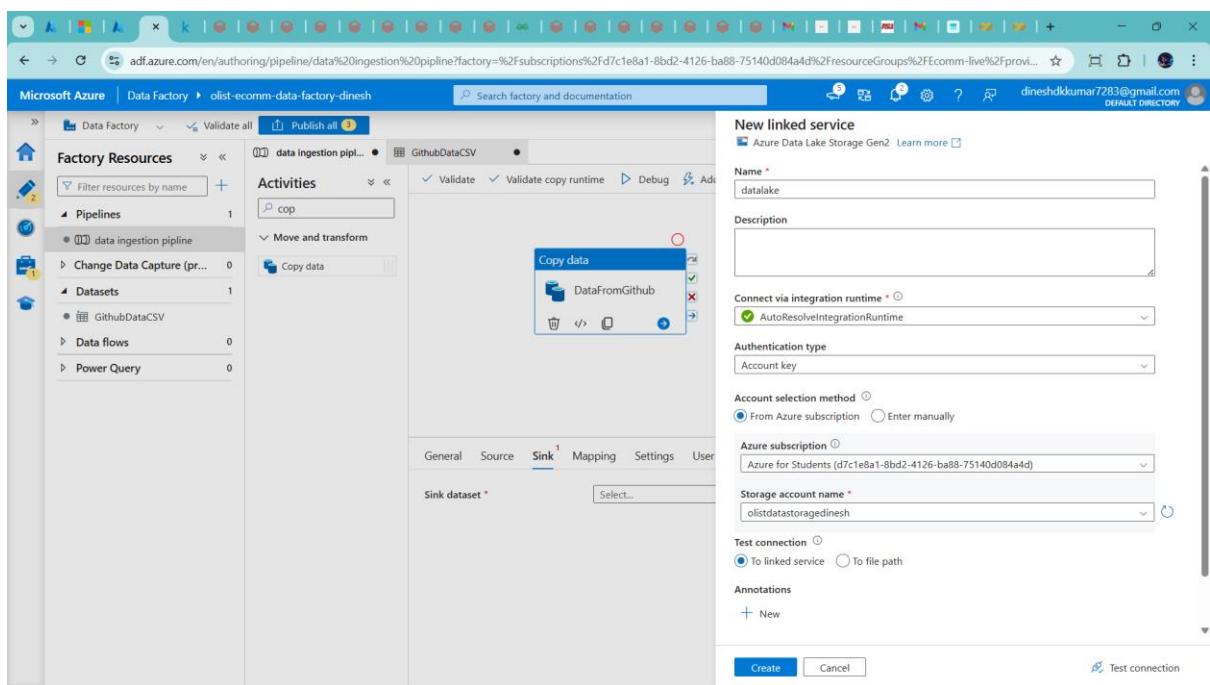
→ go to sink → click new → click data lake gen 2 → Continue

The screenshot shows the Microsoft Azure Data Factory interface for creating a new dataset. The left sidebar lists 'Factory Resources' including Pipelines, Datasets, Data flows, and Power Query. The main area shows a 'data ingestion pipeline' with a 'Copy data' activity named 'cop'. The 'Sink' tab is selected, showing a dropdown menu for 'Sink dataset'. A modal window titled 'New dataset' is open, prompting the user to 'Select a data store'. The 'All' tab is selected, displaying options like Azure Data Explorer (Kusto), Azure Data Lake Storage Gen2, Azure Database for MySQL, Azure Database for PostgreSQL, Azure Databricks Delta Lake, and Azure SQL Database. At the bottom of the modal are 'Continue' and 'Cancel' buttons.

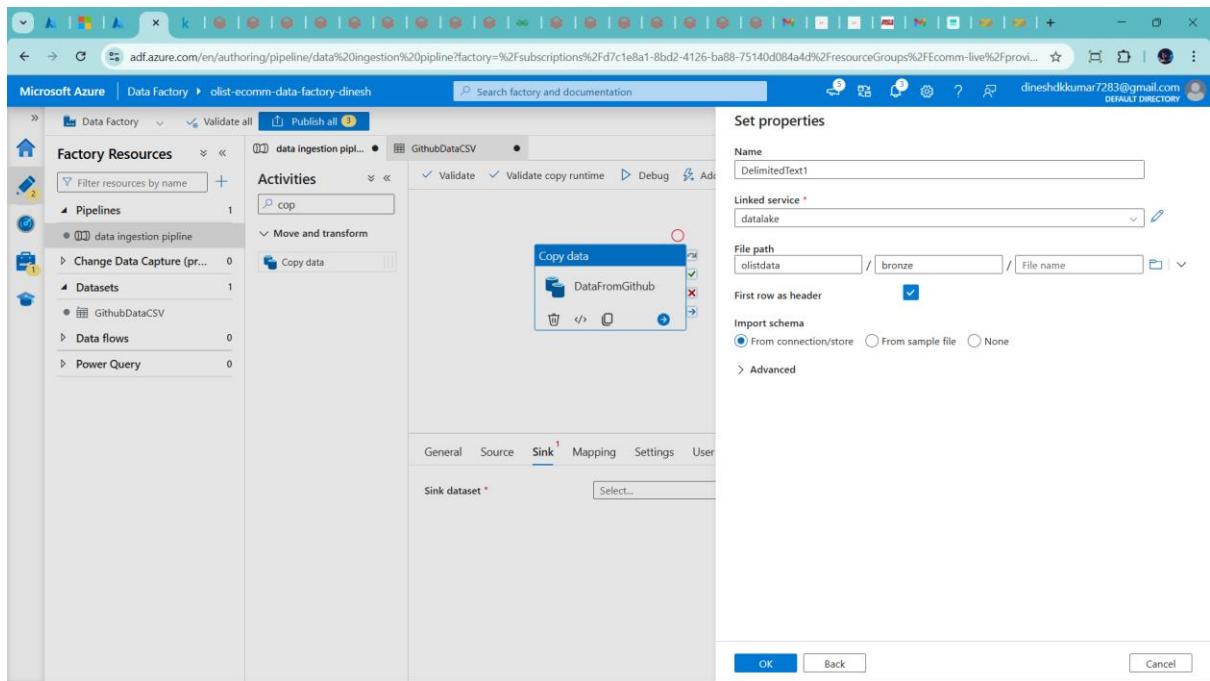
→ send to csv



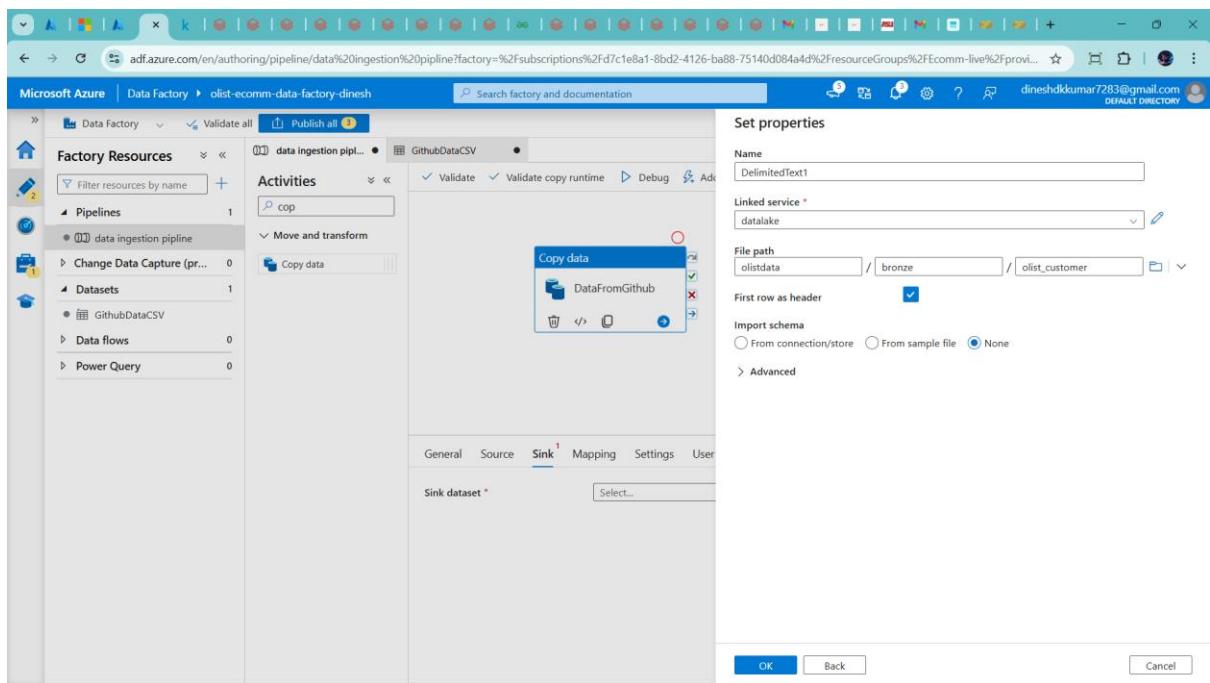
→ Click add new link service



→ Click file path icon → choose bronze folder → click ok



→ final add file name



The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'data ingestion pipeline', 'Datasets' (2), 'DelimitedText1', 'GithubDataCSV', and 'Power Query'. The main workspace displays a 'data ingestion pipeline' with a single 'Copy data' activity. The activity configuration pane at the bottom is set to the 'Sink' tab, with 'Sink dataset' set to 'DelimitedText1'. Other tabs include 'General', 'Source', 'Mapping', 'Settings', and 'User properties'. The status bar at the bottom indicates 'Pipeline run ID: 4d77b17-1160-480b-af87-febce911906b' and 'Pipeline status: Succeeded'.

→ now validated and debug

This screenshot is identical to the one above, showing the 'Copy data' activity in the pipeline. The configuration pane at the bottom shows the 'Output' tab selected, indicating a successful run. The status bar at the bottom shows the pipeline run ID and a green checkmark for the activity status.

→ hot (high cost)

Now to create parameter

→ before create delete all link service

The screenshot shows the Microsoft Azure Data Factory interface. On the left, a sidebar lists various settings like General, Connections, Source control, Security, and Author. The 'Connections' section is expanded, showing 'Linked services'. The main area is titled 'Linked services' and contains a message: 'Linked service defines the connection information to a data store or compute. Learn more'. Below this is a 'New' button and a 'Create linked service' button. A tooltip on the right says 'Will be deleted' and 'datalake (Linked service) will be deleted while publishing.'

→create new link service →https (github)

The screenshot shows the 'New linked service' configuration page for GitHub. The 'HTTP' connection type is selected. The 'Name' field is set to 'httpGitHubLinkService'. The 'Base URL' field contains 'https://raw.githubusercontent.com/dineshkumar7283/...'. Other fields include 'AutoResolveIntegrationRuntime' (selected), 'Anonymous' for authentication, and 'Enable' for server certificate validation. Buttons at the bottom include 'Create', 'Back', 'Test connection', and 'Cancel'.

→create another linked service sql

The screenshot shows the Microsoft Azure Data Factory interface. On the left, a sidebar navigation menu includes General, Connections (selected), and various other options like Integration runtimes, Microsoft Purview, and Source control. The main content area is titled 'Linked services' and displays a list of existing linked services. One entry is visible: 'httpGitHubLinkService' of type 'HTTP'. To the right, a grid of 'New linked service' options is shown under the 'Data store' tab. The grid contains nine items, each with an icon and a name: Amazon RDS for SQL Server, Azure Cosmos DB for NoSQL, Azure Database for MySQL, Azure Database for PostgreSQL, Azure SQL Database, Azure SQL Database Managed Instance, MySQL, PostgreSQL, and SQL server. Below the grid are 'Continue' and 'Cancel' buttons.

This screenshot shows the 'New linked service' configuration page for MySQL. The 'Name' field is filled with 'filesSQLDB'. The 'Description' field is empty. Under 'Connect via integration runtime', 'AutoResolveIntegrationRuntime' is selected. The 'Server name' field is empty, and the 'Port' field is set to '3306'. The 'Database name' field is empty. The 'User name' field is empty. The 'Password' field is empty and has 'Azure Key Vault' as an option. The 'SSL mode' dropdown is set to 'Drafford'. At the bottom, there are 'Create', 'Back', 'Test connection', and 'Cancel' buttons.

→fill all according in file.io

The screenshot shows the filess.io MySQL dashboard. On the left, there's a sidebar with user information (dinesh km, dineshkumar63519, 500@gmail.com, Free tier), navigation links (Databases, Standard Tier, Billing, API, Status), and a 'Get more?' section. A large MySQL logo is centered. On the right, the 'olistproject' database configuration page is displayed. It includes fields for Host (euyak.h.filess.io), Database (olistproject_tightdaily), User (olistproject_tightdaily), Port (3307), Password (redacted), MySQL URI (mysql://olistproject_tightdaily:.....@euyak.h.filess.io:3307/olistproject_tightdaily), and MySQL login command (mysql -u olistproject_tightdaily -P 3307 -p..... -h euyak.h.filess.io olistproject_tightdaily). There are also sections for 'Connection formats' and 'Backups'. At the bottom, there are 'Request Backup', 'Delete Olistproject', and 'Settings' buttons.

The screenshot shows the Microsoft Azure Data Factory 'Linked services' configuration page. The left sidebar lists various service categories like General, Connections, Source control, Author, Security, and Workflow orchestration manager. Under 'Connections', 'Linked services' is selected. The main area shows a table of linked services, with one item named 'httpGitHubLinkService' (Type: HTTP). To the right, a 'New linked service' form is open for a 'MySQL' connection. The form fields include:

- Name:** fileSQLDB
- Description:** (empty)
- Connect via integration runtime:** AutoResolveIntegrationRuntime
- Server name:** euyak.h.filess.io
- Port:** 3307
- Database name:** olistproject_tightdaily
- User name:** olistproject_tightdaily
- Password:** (redacted)
- SSL mode:** Preferred

At the bottom of the form are 'Create', 'Back', 'Test connection', and 'Cancel' buttons.

→test connection

New linked service

MySQL Learn more

Port: 3307

Database name *: olistproject_tightdaily

User name *: olistproject_tightdaily

Password *: Azure Key Vault

SSL mode: Preferred

Additional connection properties:

Annotations:

Parameters:

Create Back

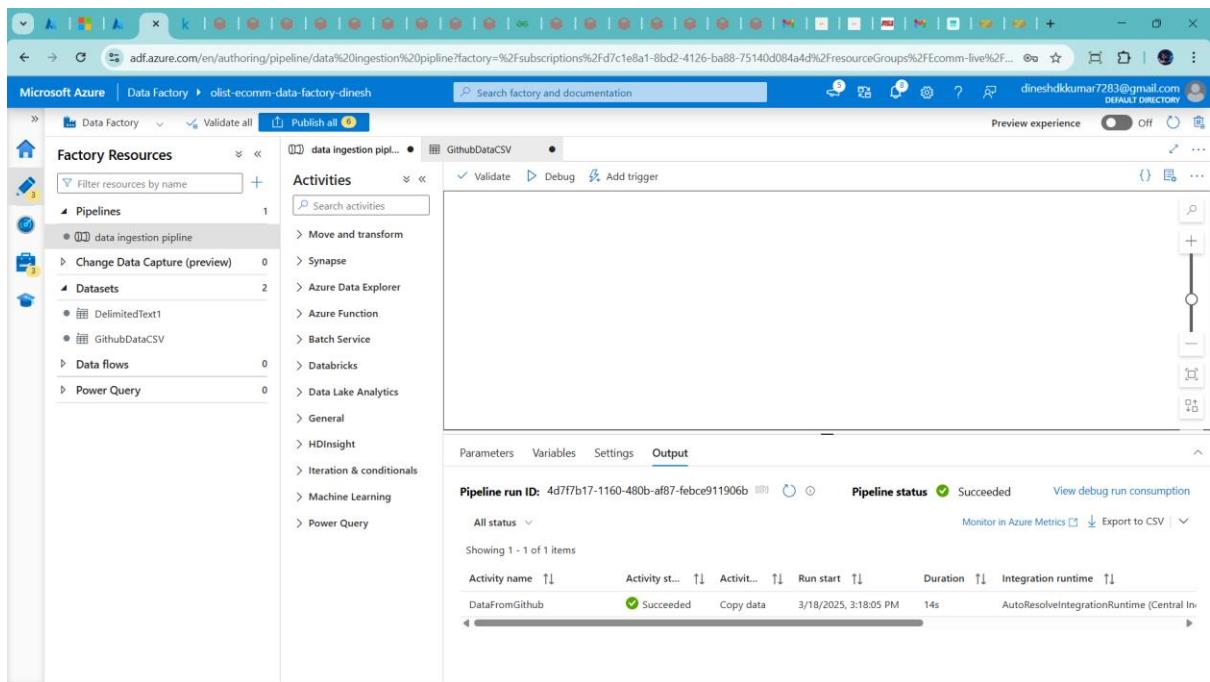
Connection successful

Test connection Cancel

→Create

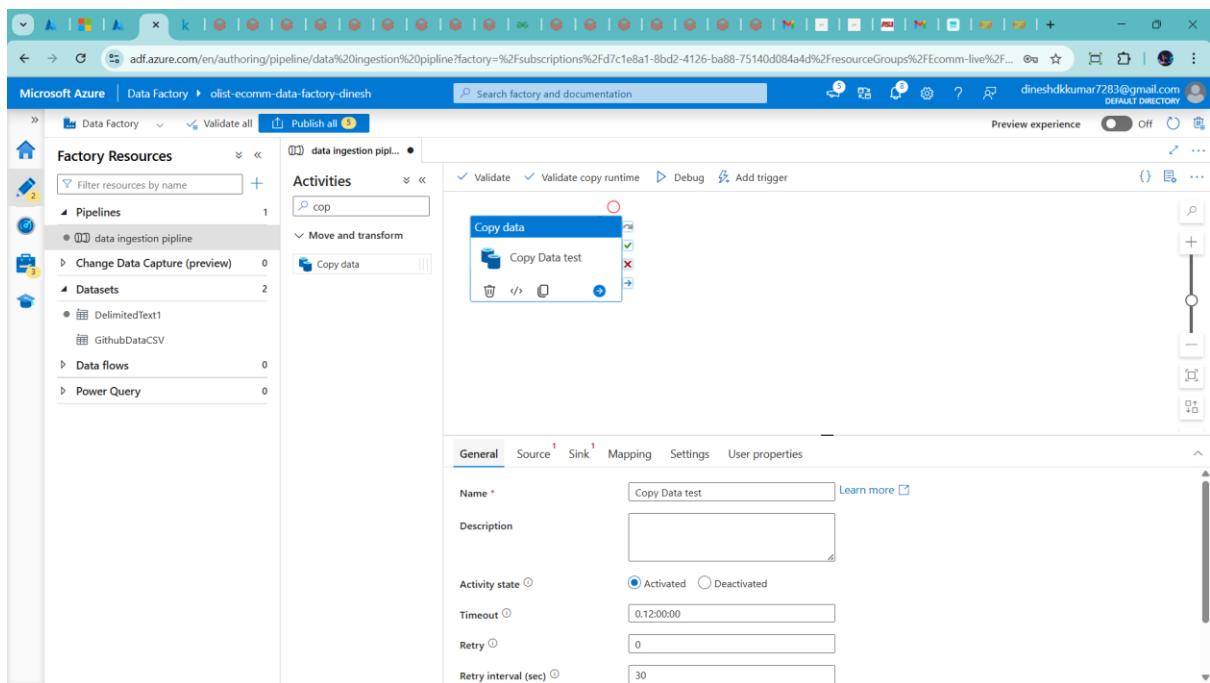
Name	Type	Related	Annotations
filesSQLDB	MySQL	0	
httpGitHubLinkService	HTTP	0	

→delete pipeline copy activity



The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists Pipelines (1), Datasets (2), and Activities (1). The main workspace displays a pipeline named 'data ingestion pipeline'. The 'Activities' section shows a single activity named 'GithubDataCSV'. Below the activities, the 'Output' tab is selected, showing a table of the pipeline run. The table has columns: Activity name, Activity state, Activit..., Run start, Duration, and Integration runtime. One row is shown: 'DataFromGitHub' with 'Succeeded' status, 'Copy data' activity, run start at '3/18/2025, 3:18:05 PM', duration '14s', and runtime 'AutoResolveIntegrationRuntime (Central In...)'.

→ drag again new one

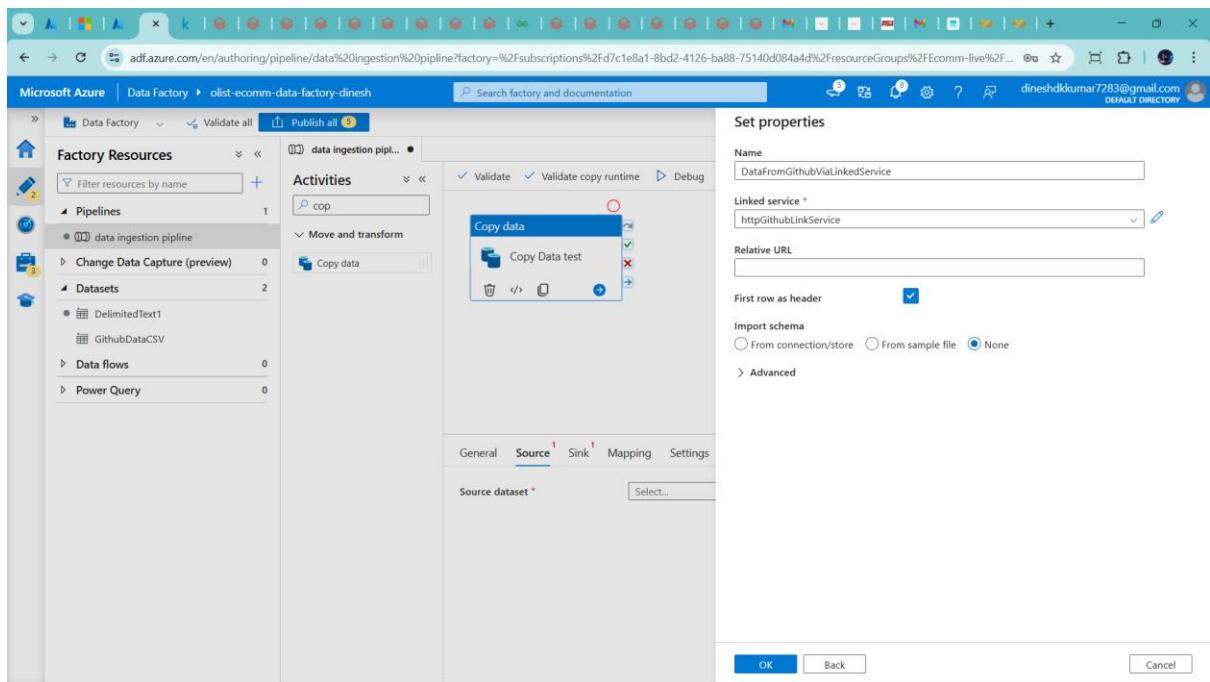


The screenshot shows the Microsoft Azure Data Factory interface. The 'Activities' section is open, and the search bar contains 'cop'. A 'Copy data' activity is selected. The configuration pane shows the 'General' tab. The 'Name' field is set to 'Copy Data test'. The 'Activity state' is set to 'Activated'. Other settings include 'Timeout' (0.12:00:00), 'Retry' (0), and 'Retry interval (sec)' (30).

→ click source and add + New

The screenshot shows the Microsoft Azure Data Factory pipeline creation interface. On the left, the 'Factory Resources' sidebar lists Pipelines, Datasets, Data flows, and Power Query. In the center, a pipeline named 'data ingestion pipeline' is selected. Under 'Activities', a 'Copy data' activity named 'Copy Data test' is being configured. The 'Source' tab is active, showing a search bar with 'htt' and a results grid with a single item: 'HTTP'. Below the grid are tabs for General, Source, Sink, Mapping, and Settings. At the bottom right are 'Continue', 'Cancel', and 'Back' buttons.

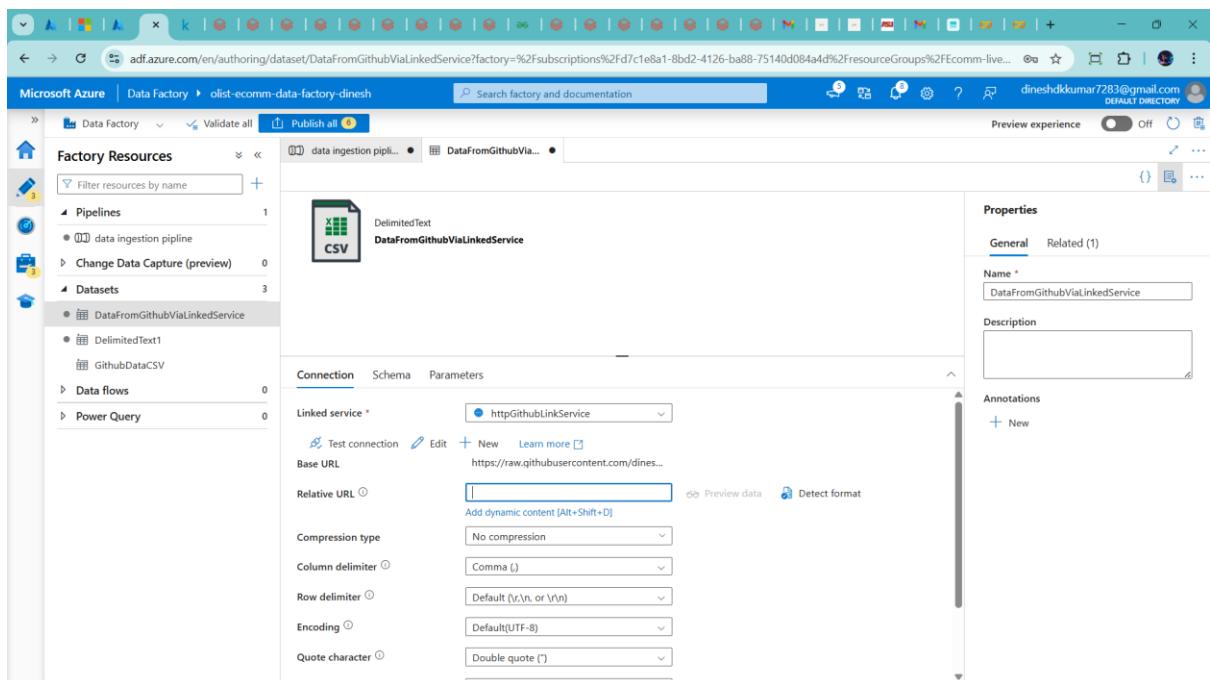
The screenshot shows the 'Select format' step in the pipeline configuration. It displays a grid of nine data formats: Avro, Binary, DelimitedText, Excel, JSON, ORC, Parquet, and XML. Each format is represented by a small icon and a label. The 'Source' tab is still active from the previous screen. At the bottom right are 'Continue', 'Back', and 'Cancel' buttons.



→ click ok

→ now add relative URL -in dynamic

→ add dynamic url



→ click parameter

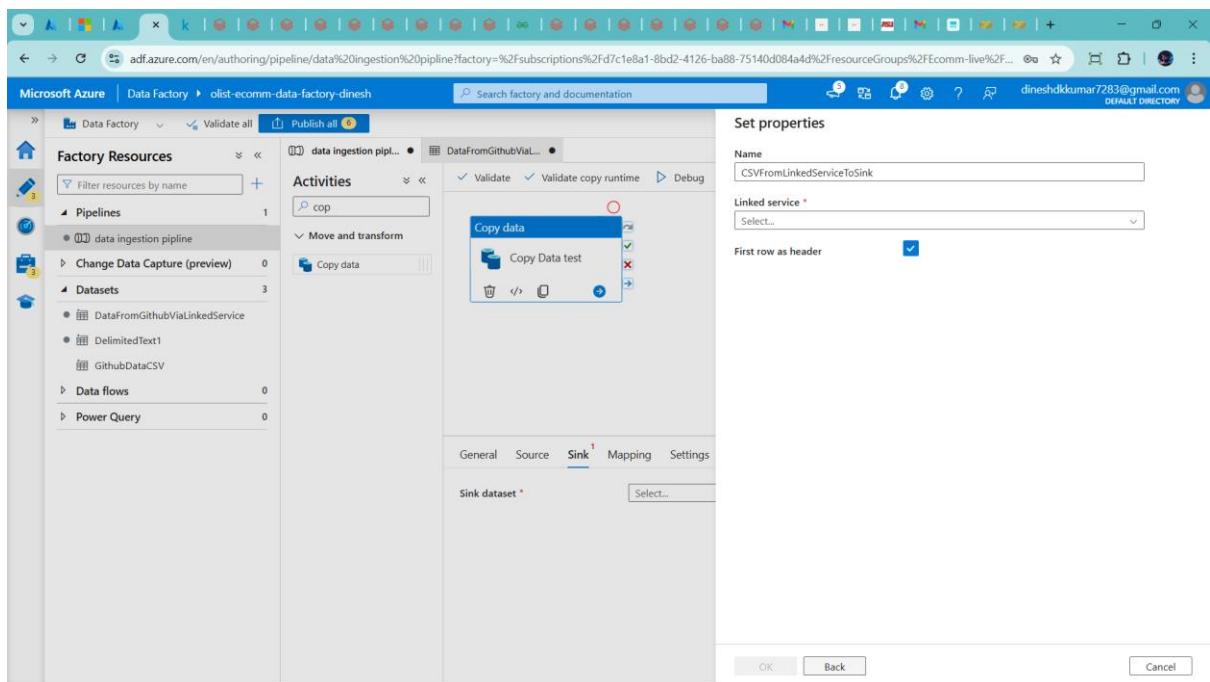
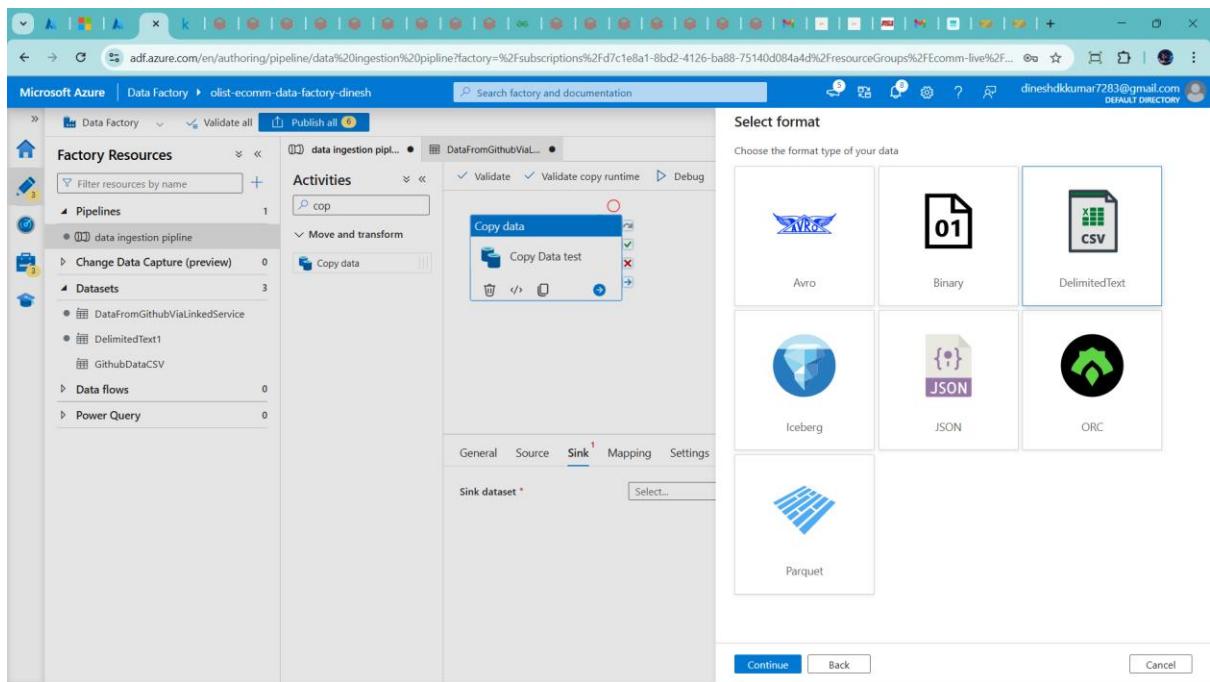
The screenshot shows the Microsoft Azure Data Factory Pipeline expression builder interface. On the left, the 'Factory Resources' sidebar lists Pipelines, Datasets, and Power Query. In the center, a pipeline named 'DataFromGithubViaLinkedService' is selected. The 'Parameters' tab is active in the top navigation bar. The main area displays connection settings for a 'httpGitHubLinkService' linked service, including fields for Base URL, Relative URL, Compression type, Column delimiter, Row delimiter, Encoding, and Quote character. A large text input field for dynamic content is present at the top right. At the bottom right are 'OK' and 'Cancel' buttons.

The screenshot shows the Microsoft Azure Data Factory Pipeline expression builder interface. The 'Parameters' tab is active. A 'New parameter' dialog is open on the right side. It has fields for 'Name' (set to 'csv_relative_url'), 'Type' (set to 'String'), and 'Default value' (empty). Below the dialog are 'Save' and 'Cancel' buttons. The central workspace shows the same pipeline configuration as the previous screenshot, including the 'Parameters' tab and connection settings for the 'httpGitHubLinkService' linked service.

The screenshot shows the 'Pipeline expression builder' dialog for a dataset named 'DataFromGithubViaLinkedService'. The 'Parameters' tab is selected, showing a search bar with 'csv_relative_url' and a list of parameters below it. The 'Functions' tab is also visible. On the left, the 'Factory Resources' sidebar shows a pipeline named 'data ingestion pipeline' and a dataset named 'DataFromGithubViaLinkedService'. The main workspace displays the dataset configuration, including its connection to 'httpGitHubLinkService', base URL 'https://raw.githubusercontent.com/dineshkumar7283/...', and various file format settings like compression type ('No compression'), column delimiter ('Comma (,),'), and quote character ('Double quote (")').

→Same as Sink →click New

The screenshot shows the 'Activities' section of a pipeline named 'data ingestion pipeline'. A 'Copy' activity is selected, and its configuration pane is open. The 'Sink' tab is active, showing a dropdown menu for 'Sink dataset *' with the placeholder 'Select...'. To the right, a 'New dataset' dialog is open, titled 'data l'. It lists data stores under the 'All' tab, with 'Azure Data Lake Storage Gen2' selected. Other tabs include 'Azure', 'Database', 'File', and 'Generic protocol'. At the bottom of the dialog are 'Continue' and 'Cancel' buttons.

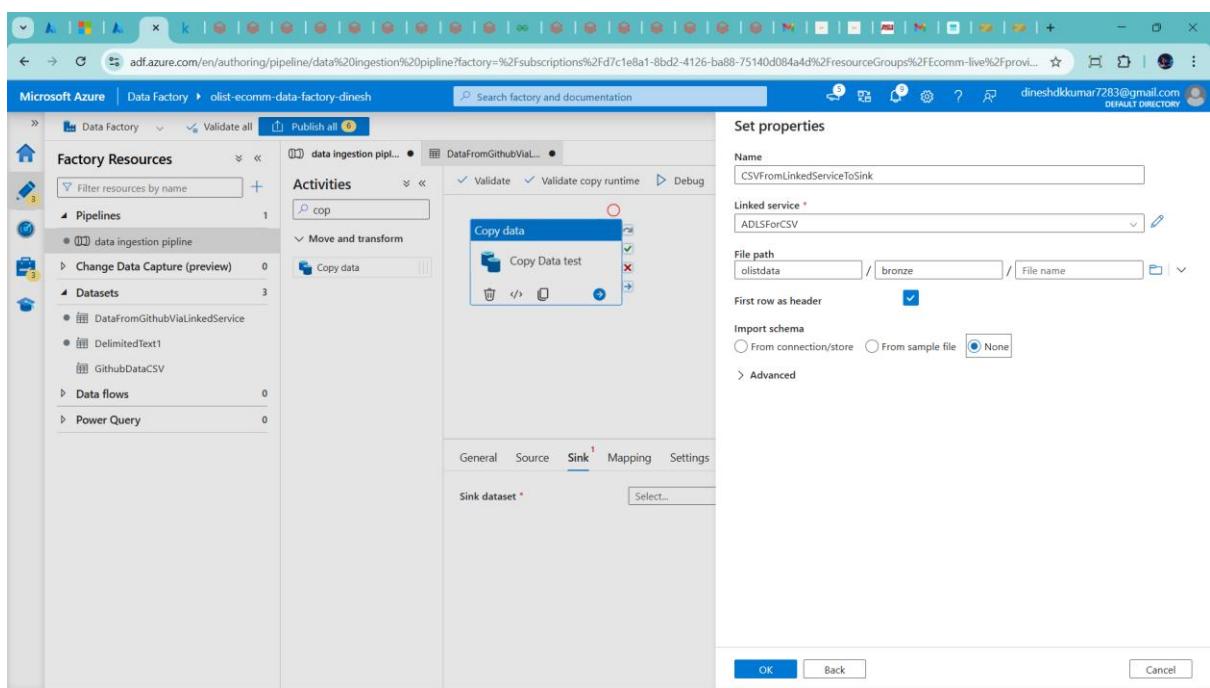
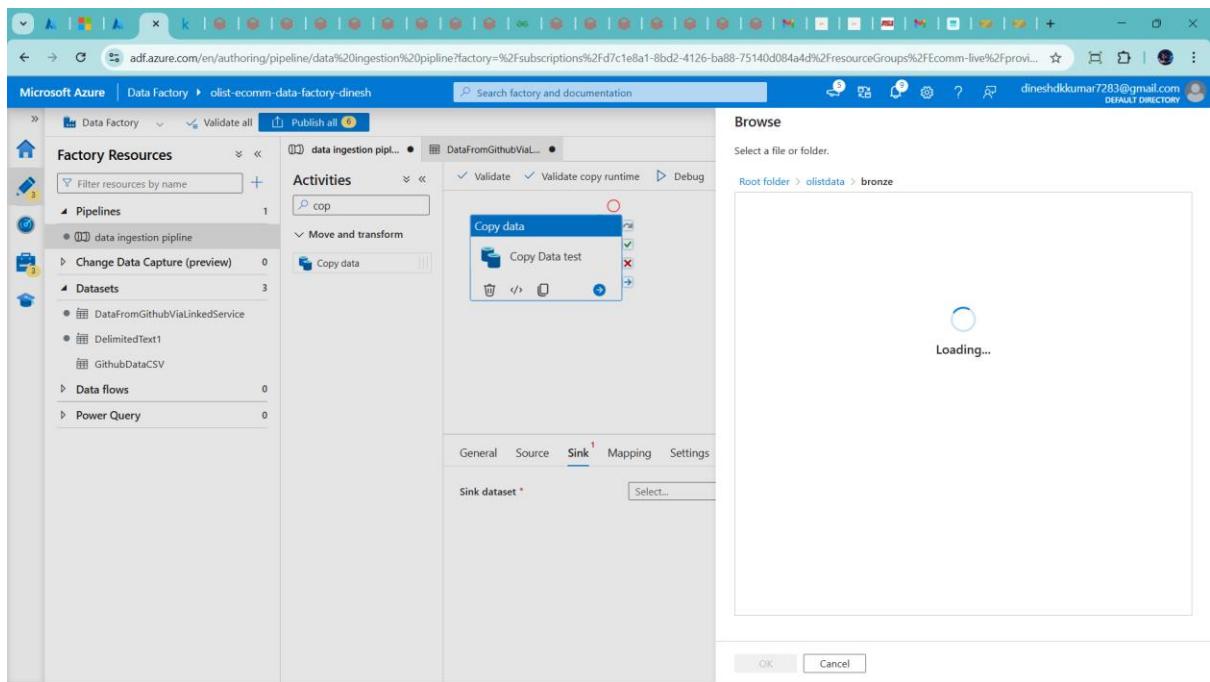


→add linked service new

The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists Pipelines, Activities, Datasets, Data flows, and Power Query. In the center, a pipeline named 'data ingestion pipeline' is selected. Under 'Activities', a 'Copy data' activity named 'Copy Data test' is being configured. The 'Sink' tab is selected, showing a dropdown for 'Sink dataset'. To the right, a 'New linked service' dialog is open for 'ADLSForCSV'. It includes fields for 'Name' (ADLSForCSV), 'Description', 'Connect via integration runtime' (AutoResolveIntegrationRuntime), 'Authentication type' (Account key), 'Account selection method' (From Azure subscription), 'Azure subscription' (Azure for Students), 'Storage account name' (olistdatastoragedinsh), and 'Test connection' (To linked service). At the bottom are 'Create' and 'Cancel' buttons.

This screenshot is similar to the one above, but the 'New linked service' dialog has been closed, and a success message 'Successfully created' is displayed above the 'Linked service' section. The 'Linked service' field now contains 'ADLSForCSV'. The rest of the interface remains the same, showing the pipeline editor with the 'data ingestion pipeline' and its 'Copy data' activity.

→choose bronze folder



The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists Pipelines (1), Datasets (4), and other components. In the center, a pipeline named 'data ingestion pipeline' is selected, showing a single 'Copy data' activity named 'Copy data test'. The 'Sink' tab is active in the configuration pane below. The sink dataset is set to 'CSVFromLinkedServiceToSink'. Other settings include 'Copy behavior' (Select...), 'Max concurrent connections' (empty), 'Block size (MB)' (empty), 'Metadata' (New), and 'Quote all text' (checked). The preview experience toggle is off.

→ add filename as dynamic

The screenshot shows the Microsoft Azure Data Factory dataset editor for 'CSVFromLinkedServiceToSink'. The 'Properties' pane on the right shows the dataset name as 'CSVFromLinkedServiceToSink'. The 'Connection' tab in the center shows the linked service is 'ADLSForCSV'. The 'File path' field contains the value 'olistdata/bronze' followed by a dynamic placeholder '/filename'. A tooltip indicates 'Add dynamic content [Alt+Shift+D]'. Other connection parameters include 'Compression type' (No compression), 'Column delimiter' (Comma (,), 'Row delimiter' (Default (\r\n or \n\r)), 'Encoding' (Default(UTF-8)), 'Quote character' (Double quote (')), and 'Escape character' (Backslash (\)).

→ click add dynamin and create new parameter

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists Pipelines, Datasets, Data flows, and Power Query. Under 'Datasets', 'CSVFromLinkedServiceToSink' is selected. The main panel displays the 'CSVFromLinkedServiceToSink' dataset configuration. The 'Connection' tab is active, showing 'Linked service' set to 'ADLSForCSV'. The 'File path' field contains 'olistdata / bronze'. The 'Parameters' tab is also visible. A 'New parameter' dialog is open on the right, prompting for a name ('file_name'), type ('String'), and default value. The 'Save' button is at the bottom right of the dialog.

This screenshot shows the same dataset configuration as the previous one, but with a parameter reference added to the file path. In the 'File path' field, the value 'bronze' is replaced by '@dataset().file_name'. The 'Properties' pane on the right shows the dataset's name is 'CSVFromLinkedServiceToSink'. The 'Annotations' section is also visible.

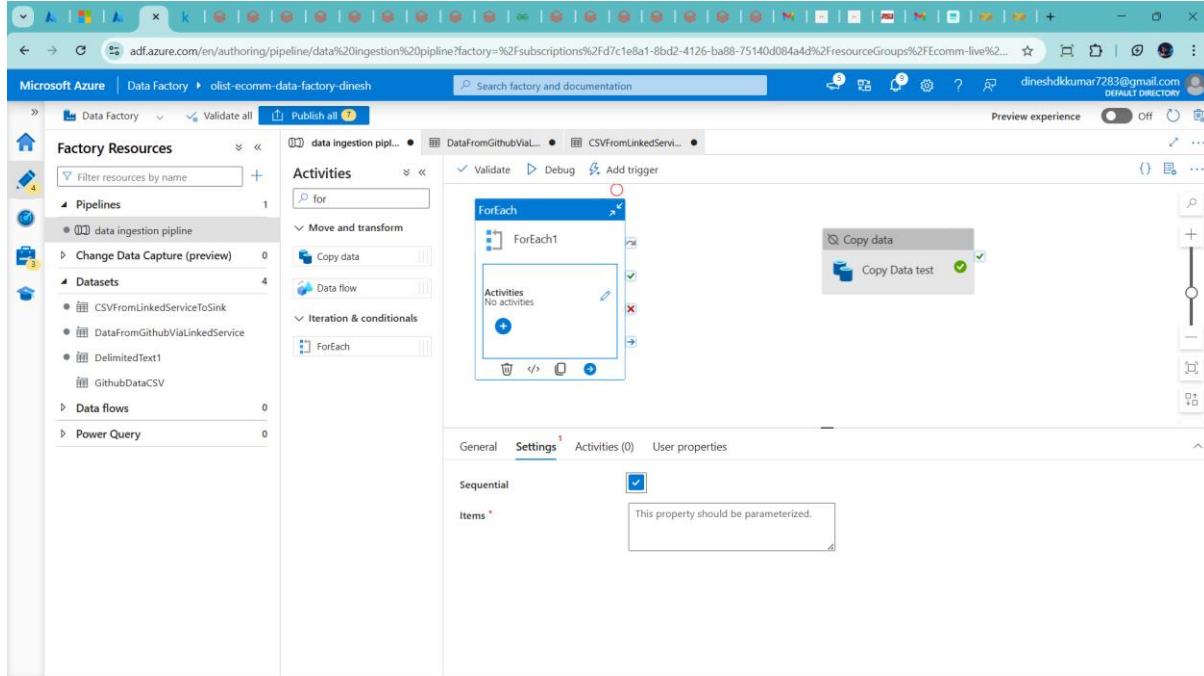
Now we For Each loop

The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (4), 'Data flows' (0), and 'Power Query' (0). The main workspace displays a pipeline named 'data ingestion pipl...' under the 'data ingestion pipeline'. The pipeline contains a 'ForEach' activity named 'ForEach1' which has an 'Activities' sub-section. A 'Copy data' activity named 'Copy data test' is connected to 'ForEach1'. The pipeline status bar indicates 'Validate all' and 'Publish all' buttons, along with a 'Preview experience' toggle switch set to 'Off'.

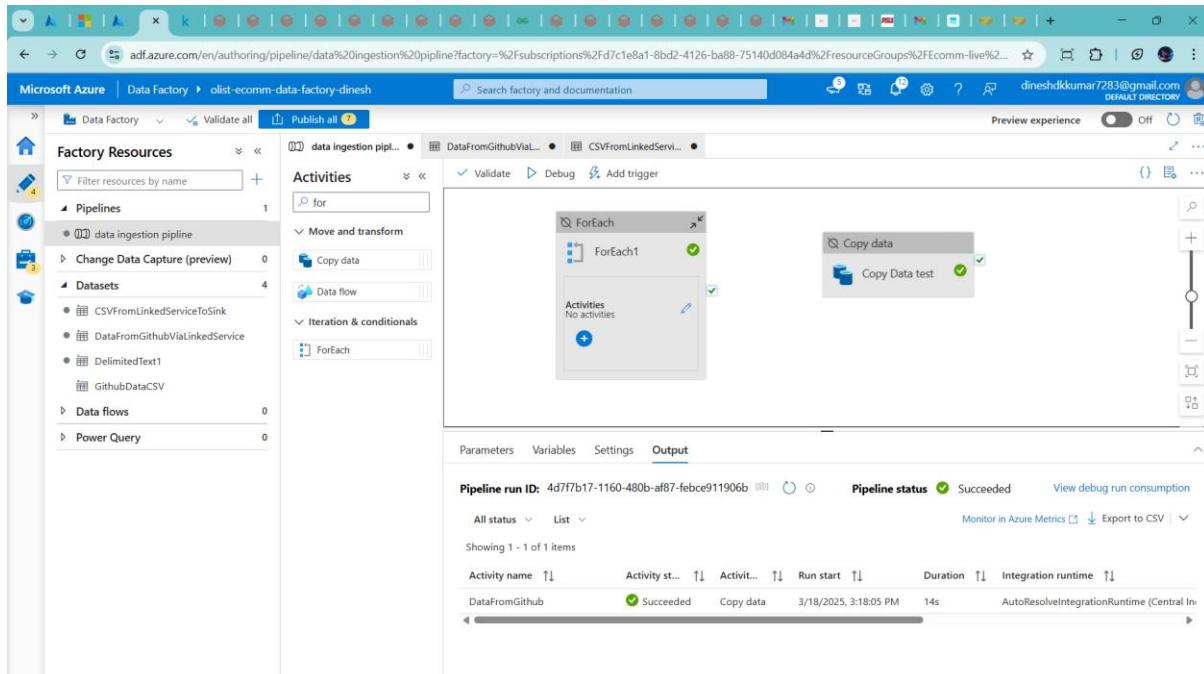
→enable sequincal

Sequencial →line by line

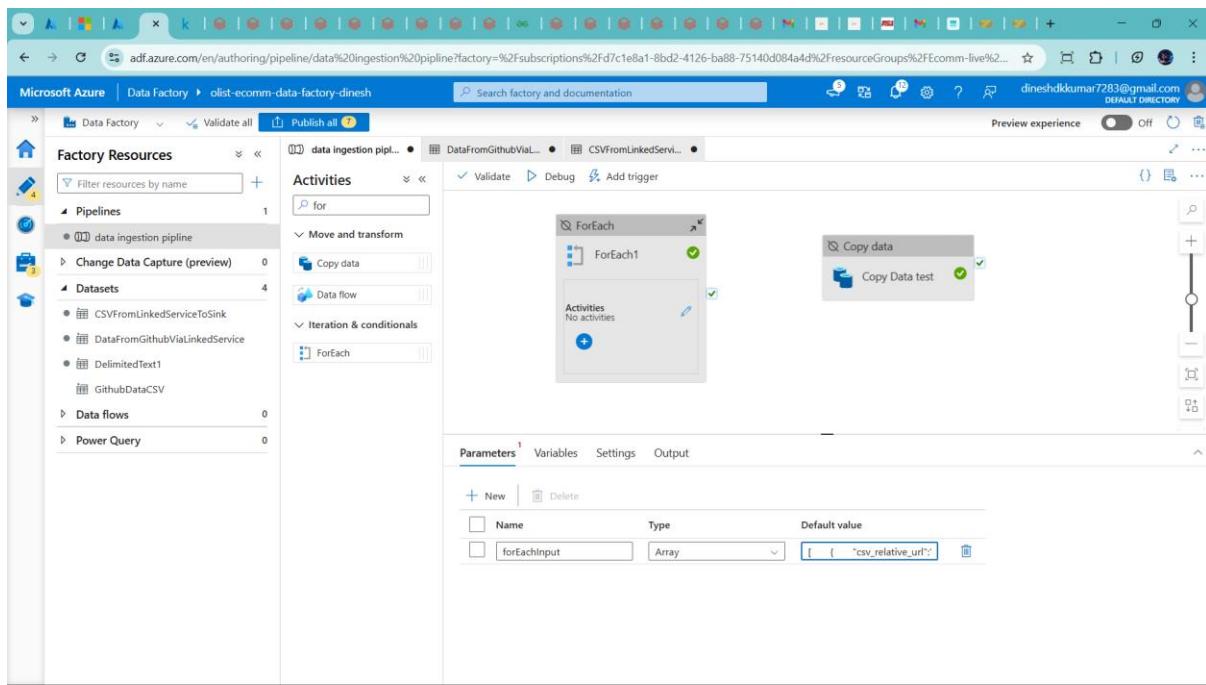
Batch count → split to run parallel



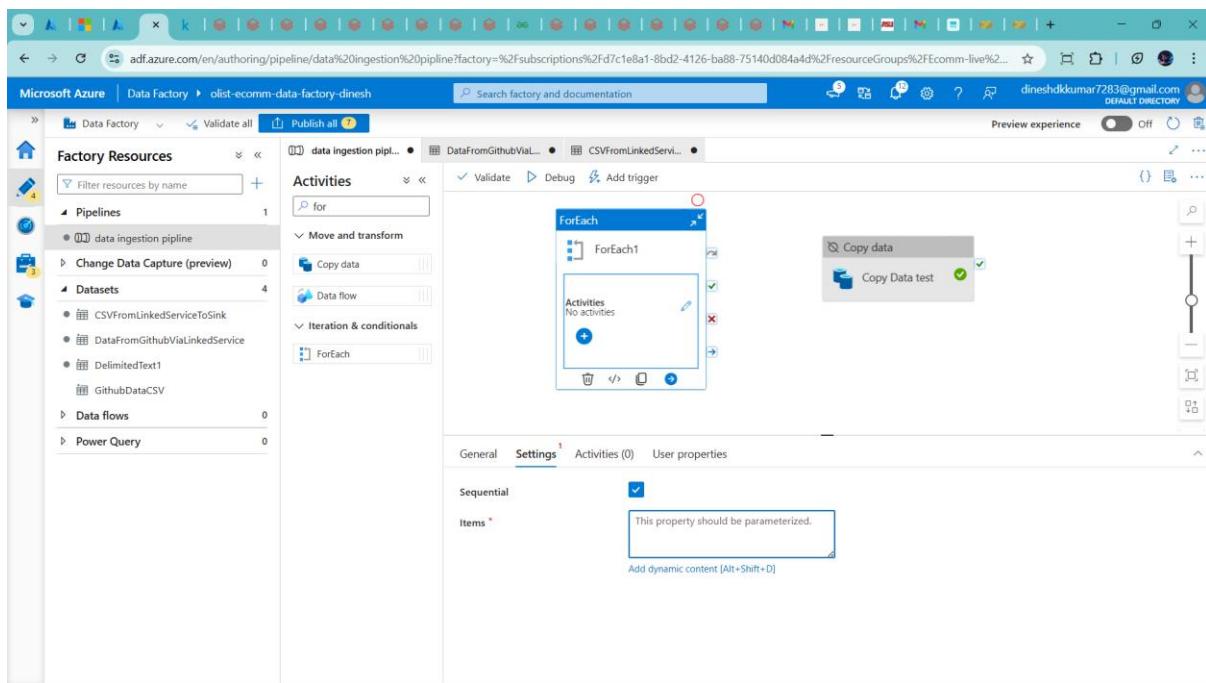
→click outside of activity

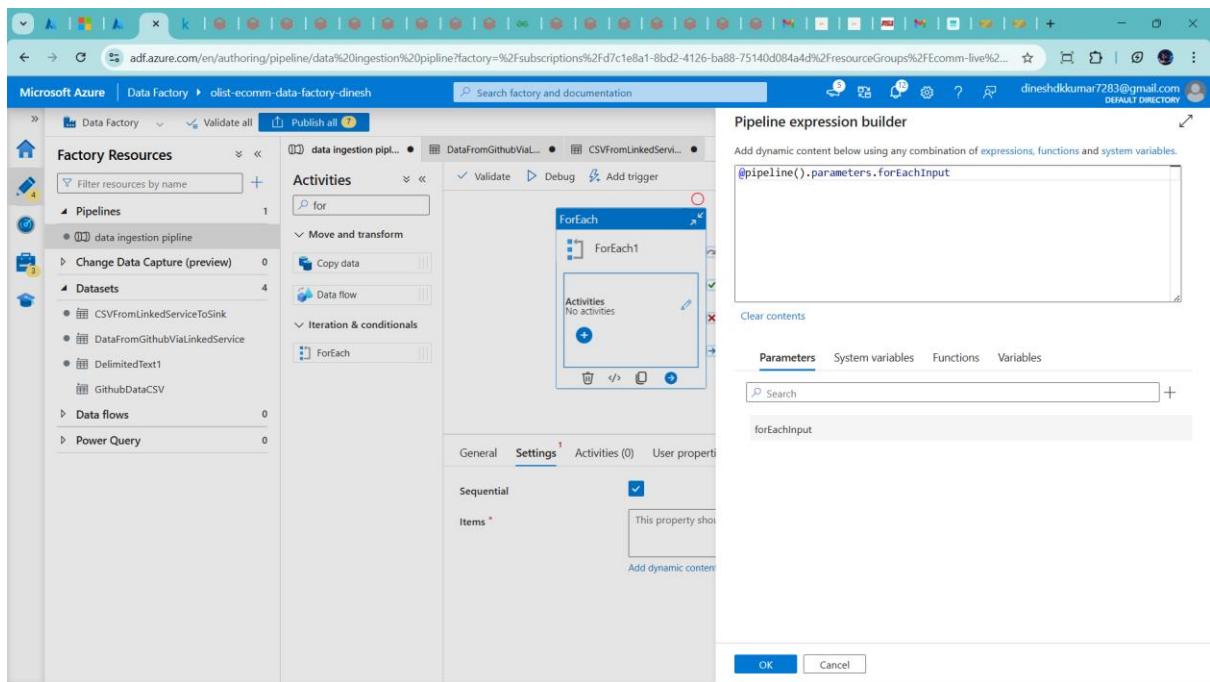


Click parameter

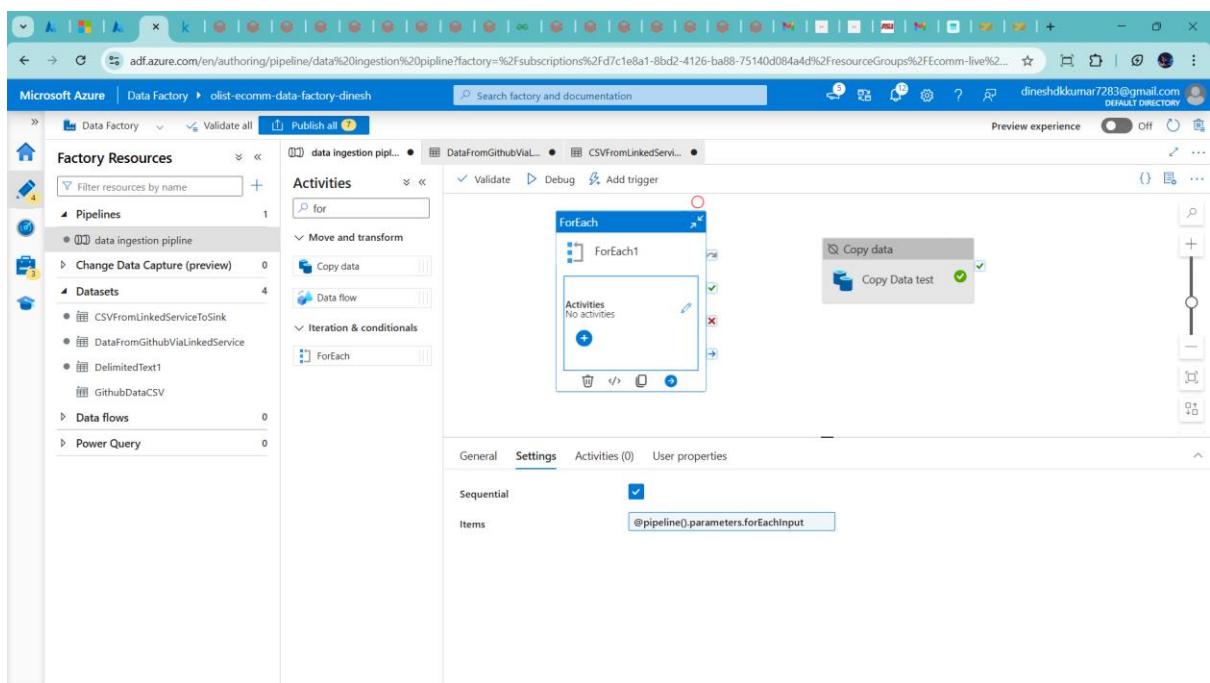


→click setting →add dynamic content





→ click ok



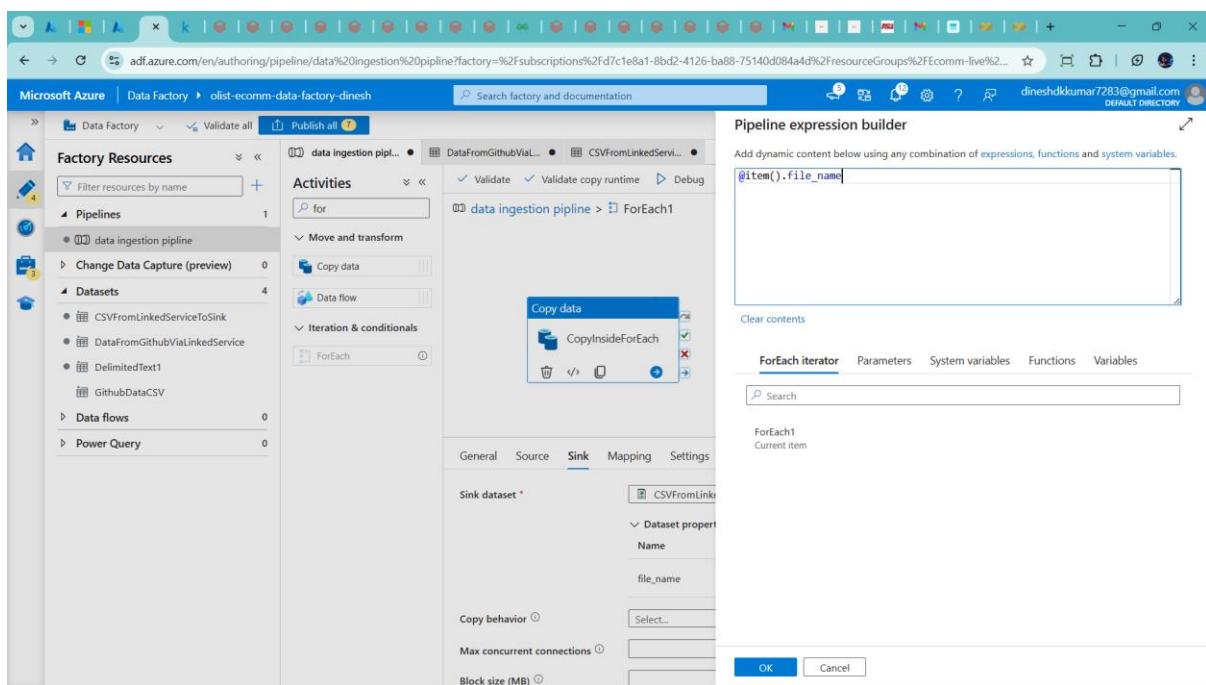
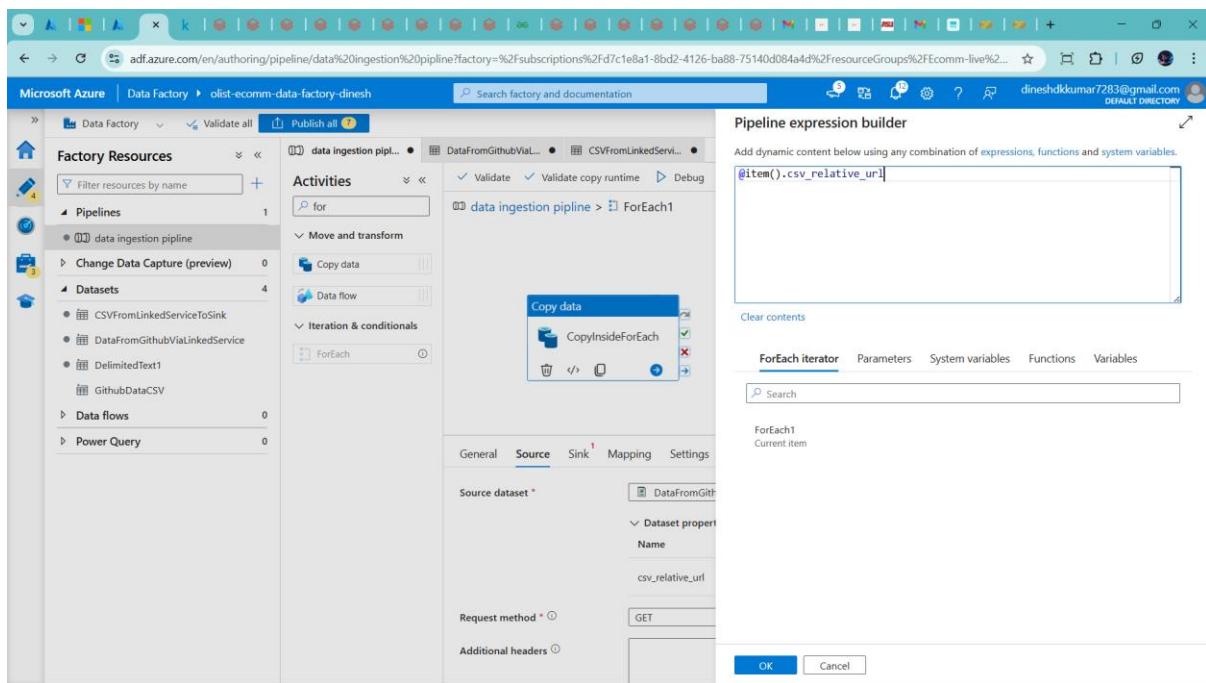
→

→ Click for each edit pencil → drag copy data

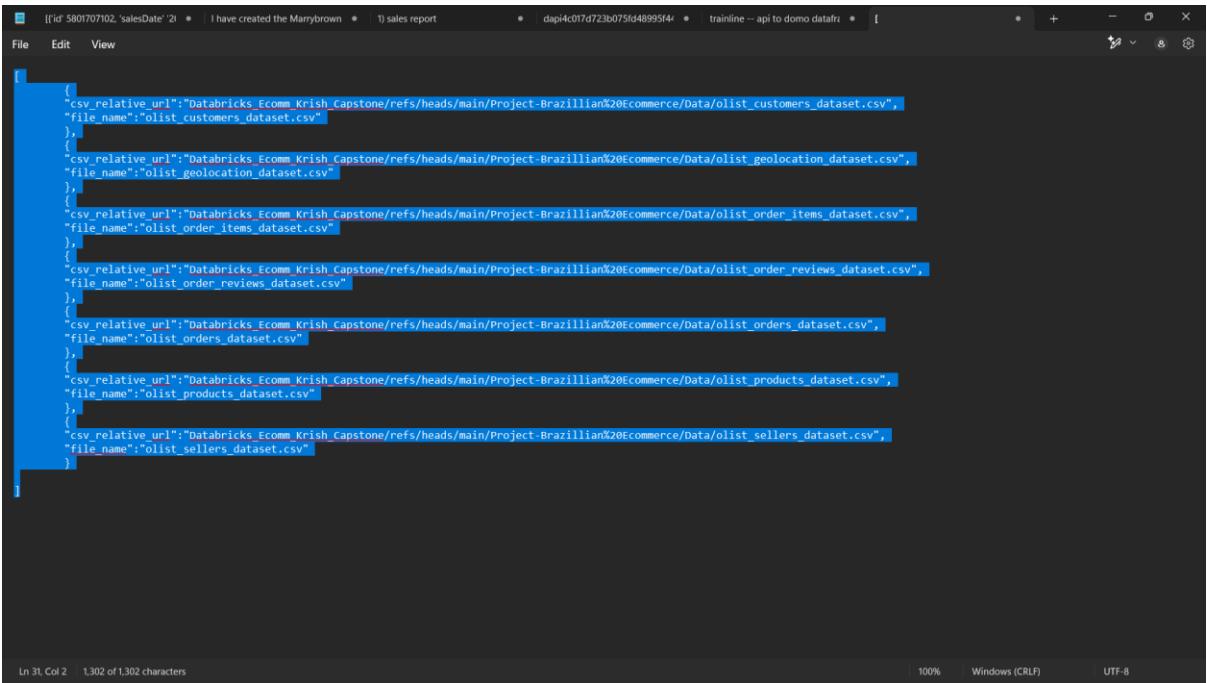
The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists Pipelines (1), Datasets (4), and Data flows (0). The main workspace displays a 'data ingestion pipeline > ForEach1' activity. Inside this activity, there is a 'Copy data' component. Below the component, the 'General' tab of the properties pane is selected, showing fields for Name (Copy data1), Activity state (Activated), Timeout (0.12:00:00), Retry (0), and Retry interval (sec) (30).

→ add source and sink link same as above

The screenshot shows the Microsoft Azure Data Factory pipeline editor with the 'Source' tab selected for the 'Copy data' activity. Under 'Source dataset', it is set to 'DataFromGithubViaLinkedService'. The 'Dataset properties' section contains a single entry: 'Name' (csv_relative_url) and 'Value' (@item().csv_relative_url), both of which are highlighted in blue. Other tabs visible include General, Sink, Mapping, Settings, and User properties.



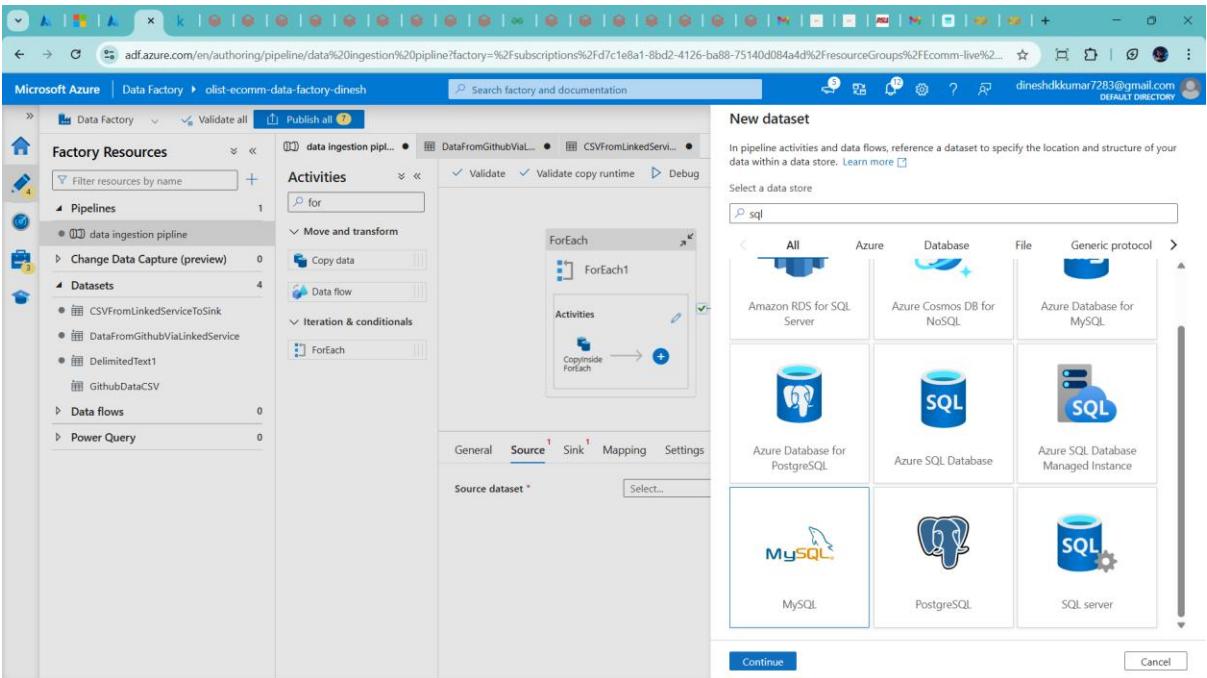
→correct



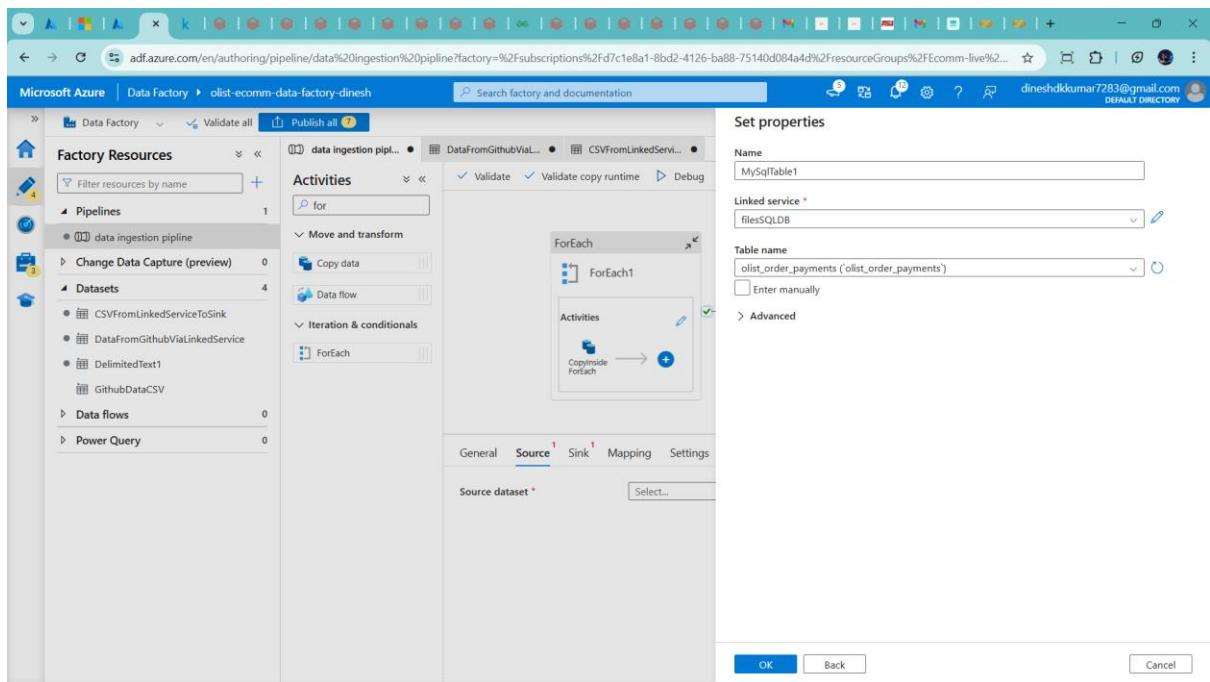
```
[{"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_customers_dataset.csv", "file_name": "olist_customers_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_geolocation_dataset.csv", "file_name": "olist_geolocation_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_order_items_dataset.csv", "file_name": "olist_order_items_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_order_reviews_dataset.csv", "file_name": "olist_order_reviews_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_orders_dataset.csv", "file_name": "olist_orders_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_products_dataset.csv", "file_name": "olist_products_dataset.csv"}, {"csv_relative_url": "Databricks_Ecomm_Krish_Capstone/refs/heads/main/Project-Brazillian%20Ecommerce/Data/olist_sellers_dataset.csv", "file_name": "olist_sellers_dataset.csv"}]
```

In 31 Col 2 1,302 of 1,302 characters 100% Windows (CRLF) UTF-8

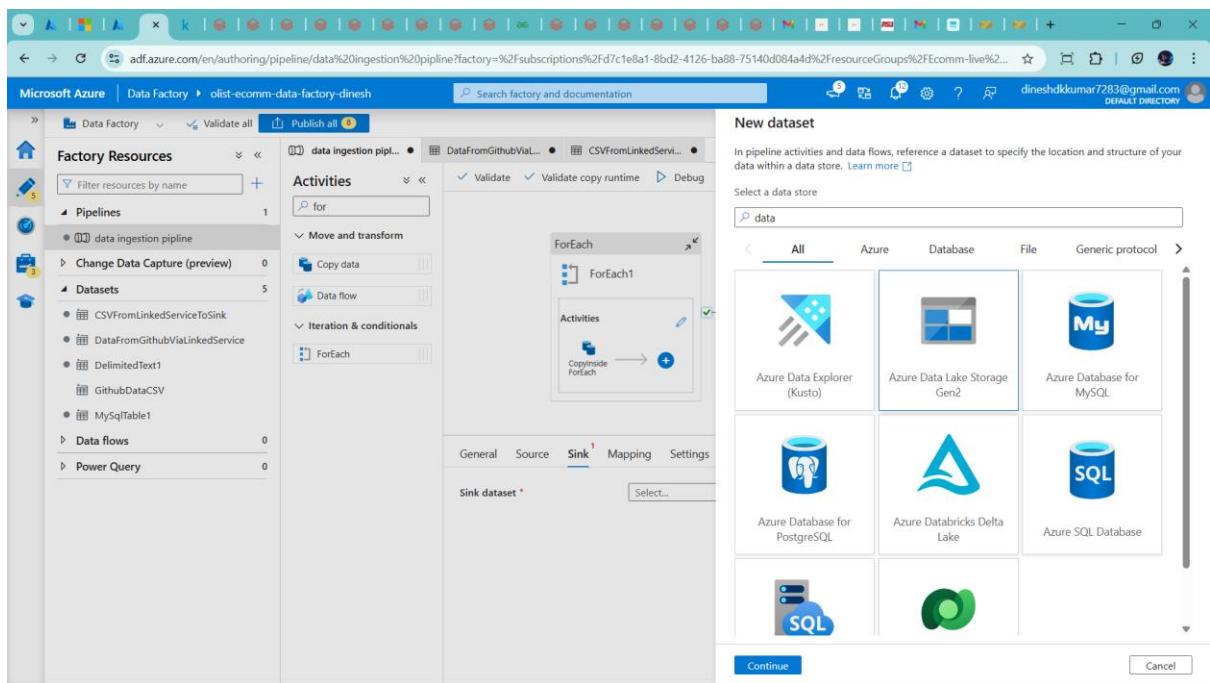
→Now add SQL source in this copy activity

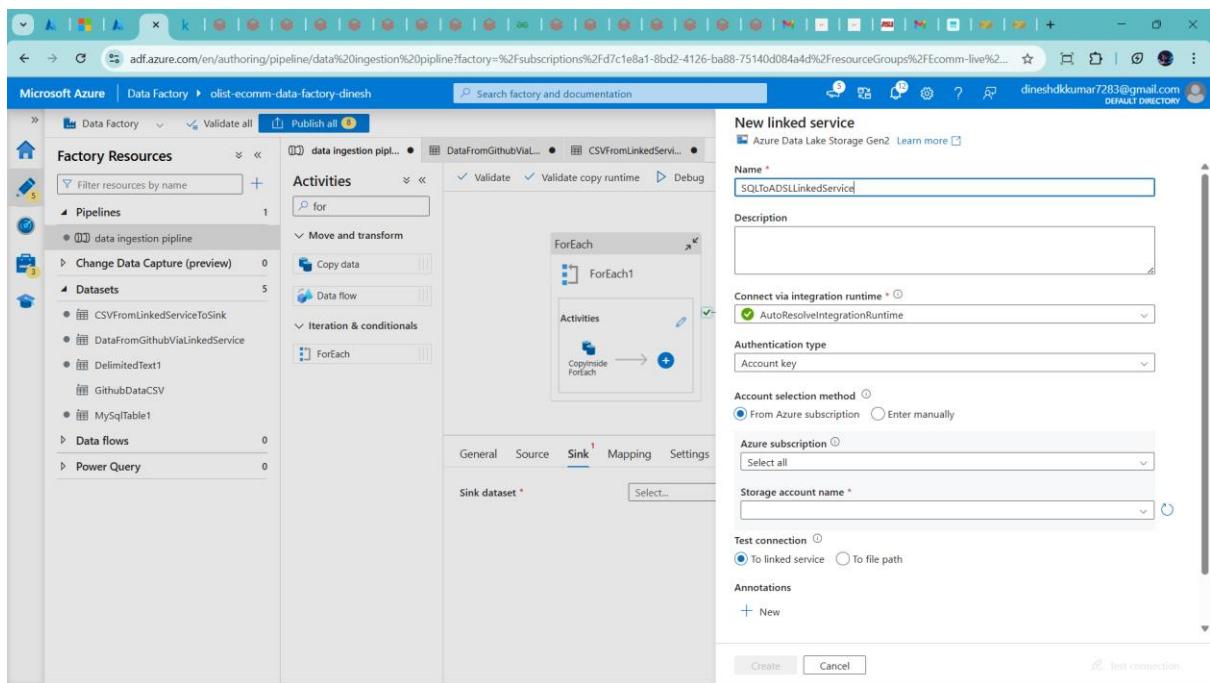
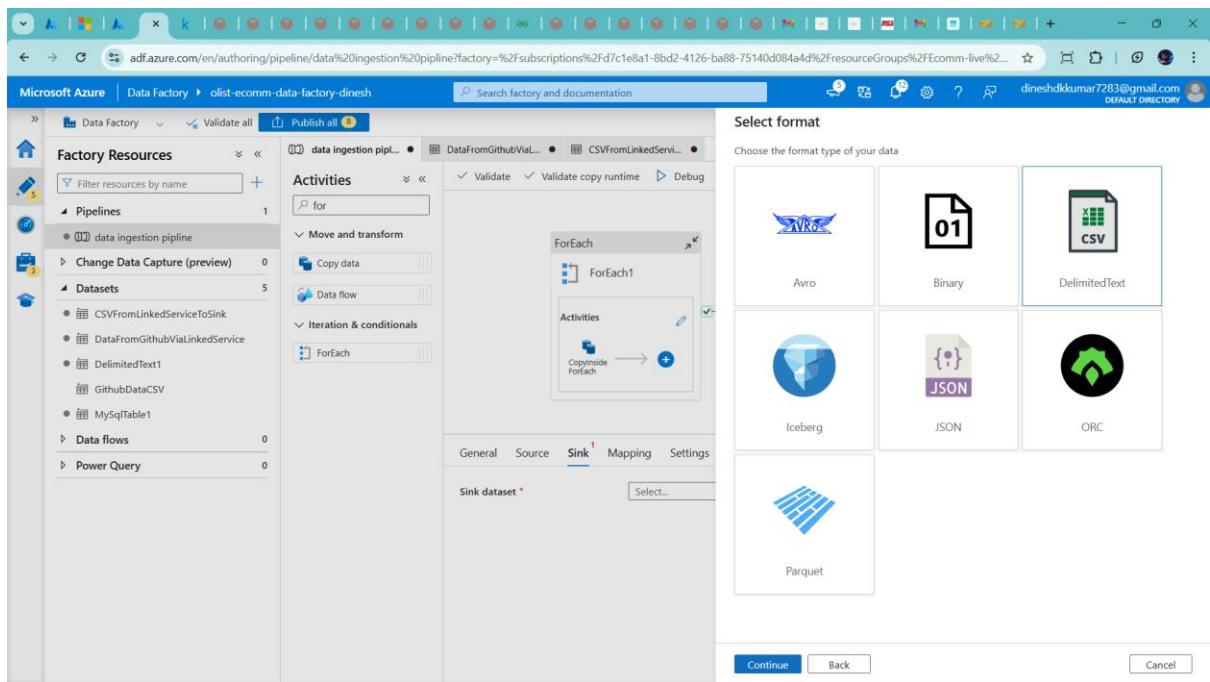


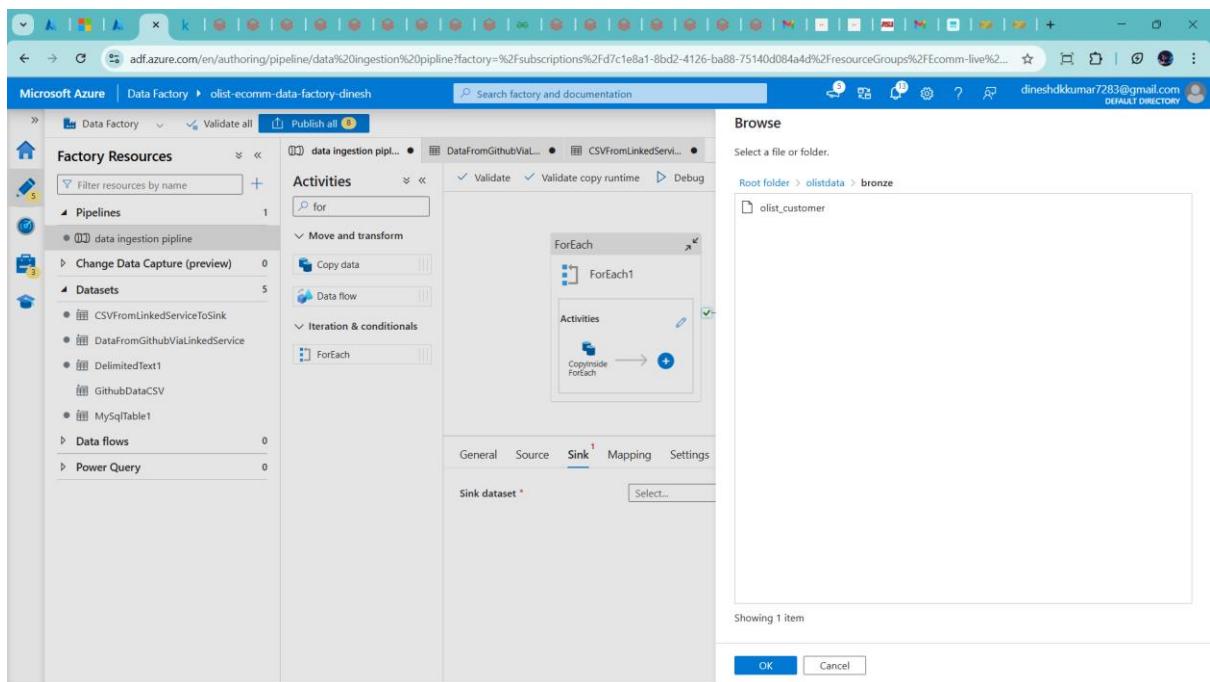
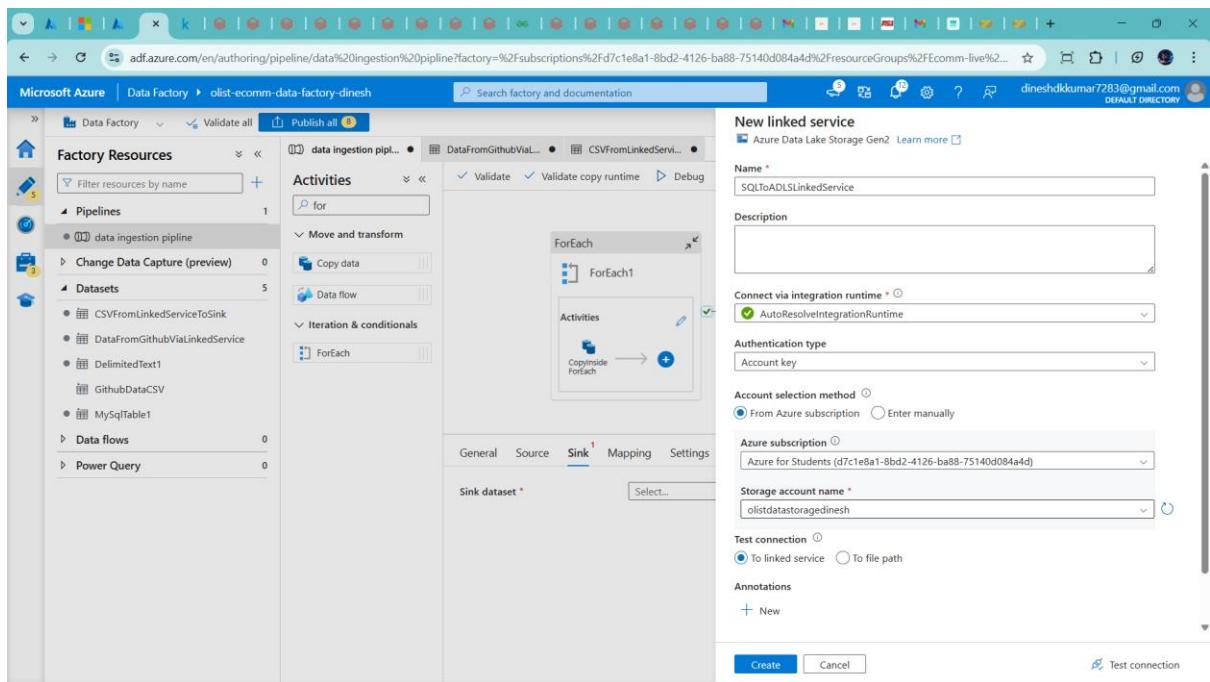
The screenshot shows the Microsoft Azure Data Factory pipeline designer. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (4), 'Data flows' (0), and 'Power Query' (0). In the center, a pipeline named 'data ingestion pipeline' is selected. A 'Copy data' activity is currently being configured. The 'Source' tab is active, showing a 'Select...' button. To the right, a 'New dataset' dialog is open, prompting the user to 'Select a data store'. The 'sql' search term has been entered, and a grid of database options is displayed, including 'All', 'Azure', 'Database', 'File', and 'Generic protocol'. Specific options shown include 'Amazon RDS for SQL Server', 'Azure Cosmos DB for NoSQL', 'Azure Database for MySQL', 'Azure Database for PostgreSQL', 'Azure SQL Database Managed Instance', 'MySQL', 'PostgreSQL', and 'SQL server'. At the bottom of the dialog are 'Continue' and 'Cancel' buttons.

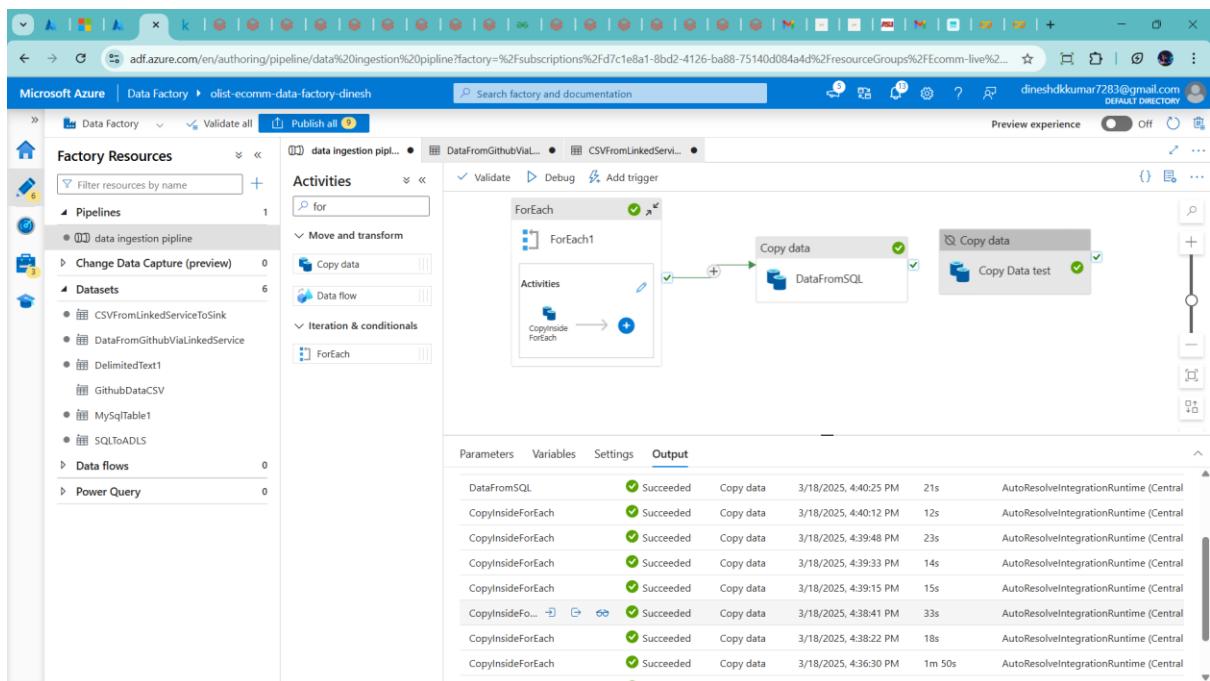
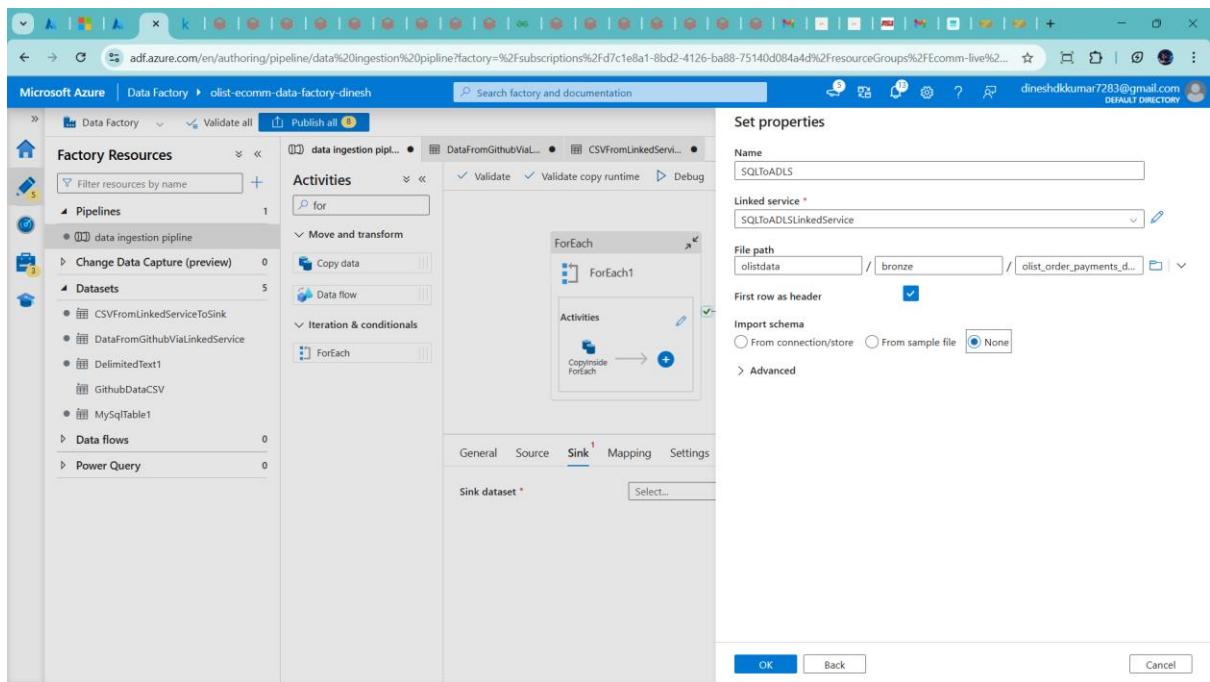


→in sink

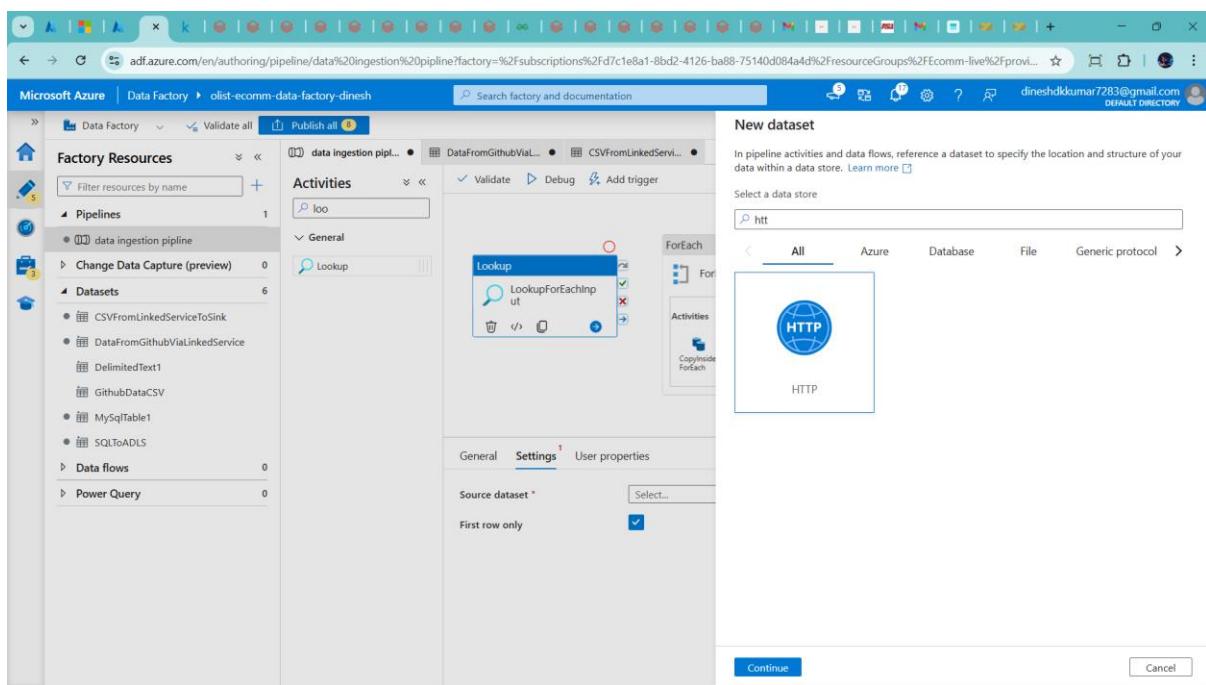
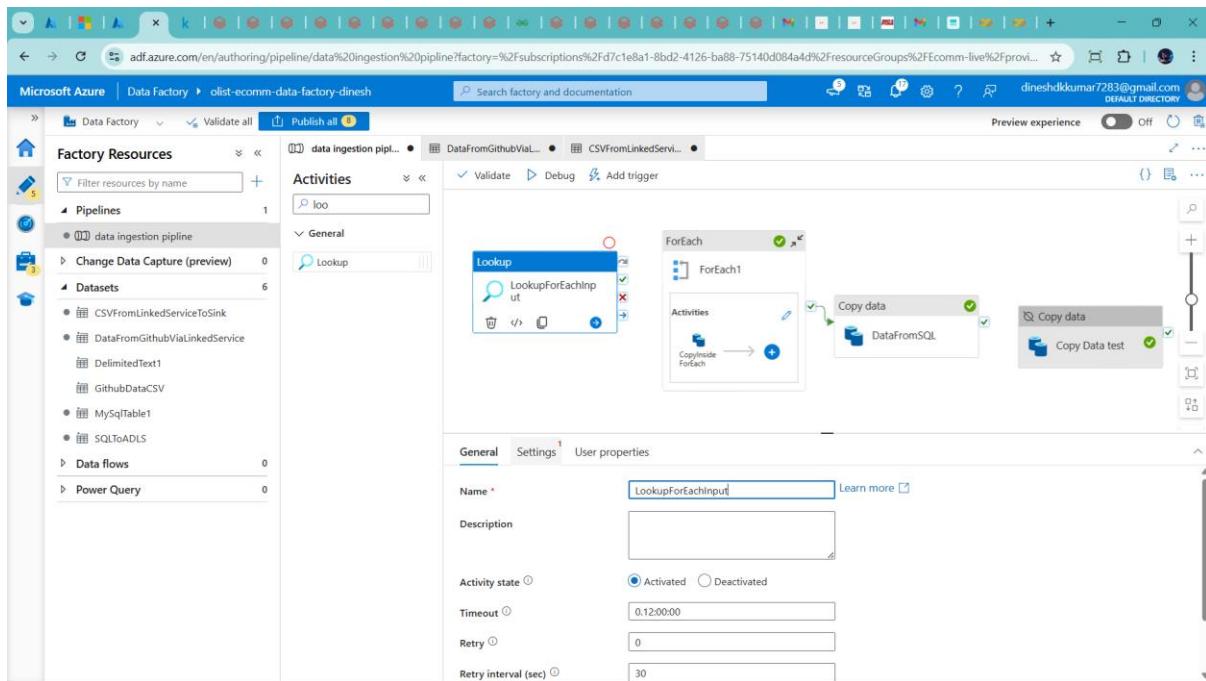


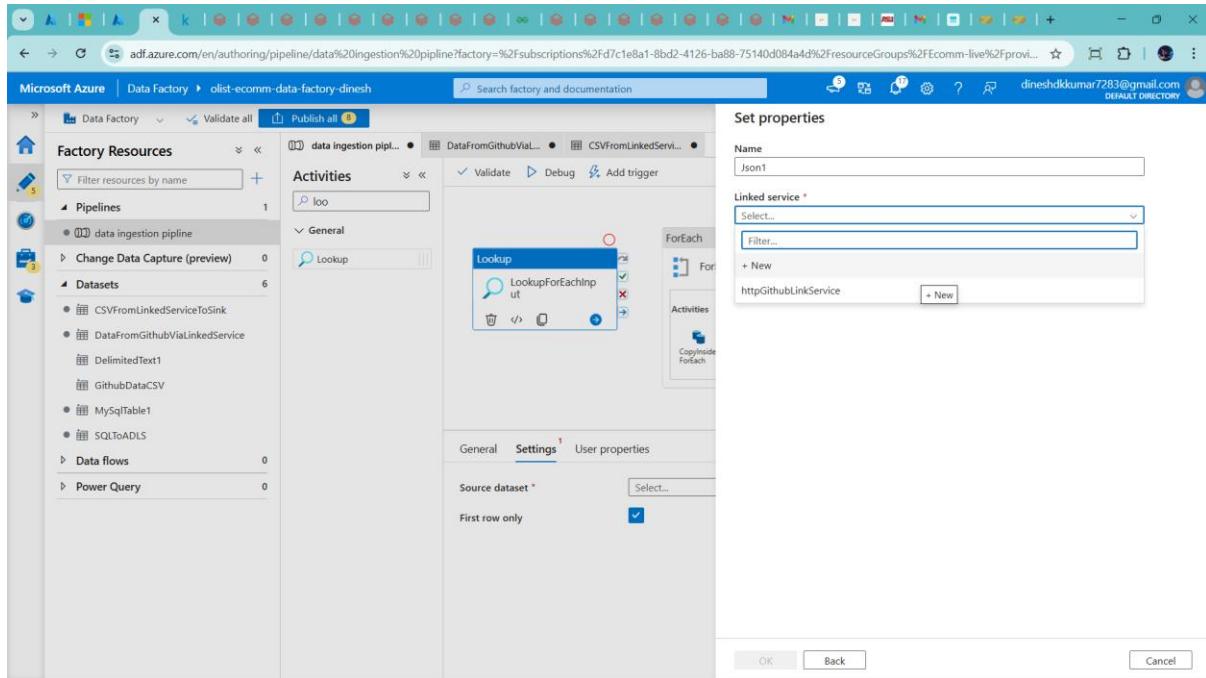
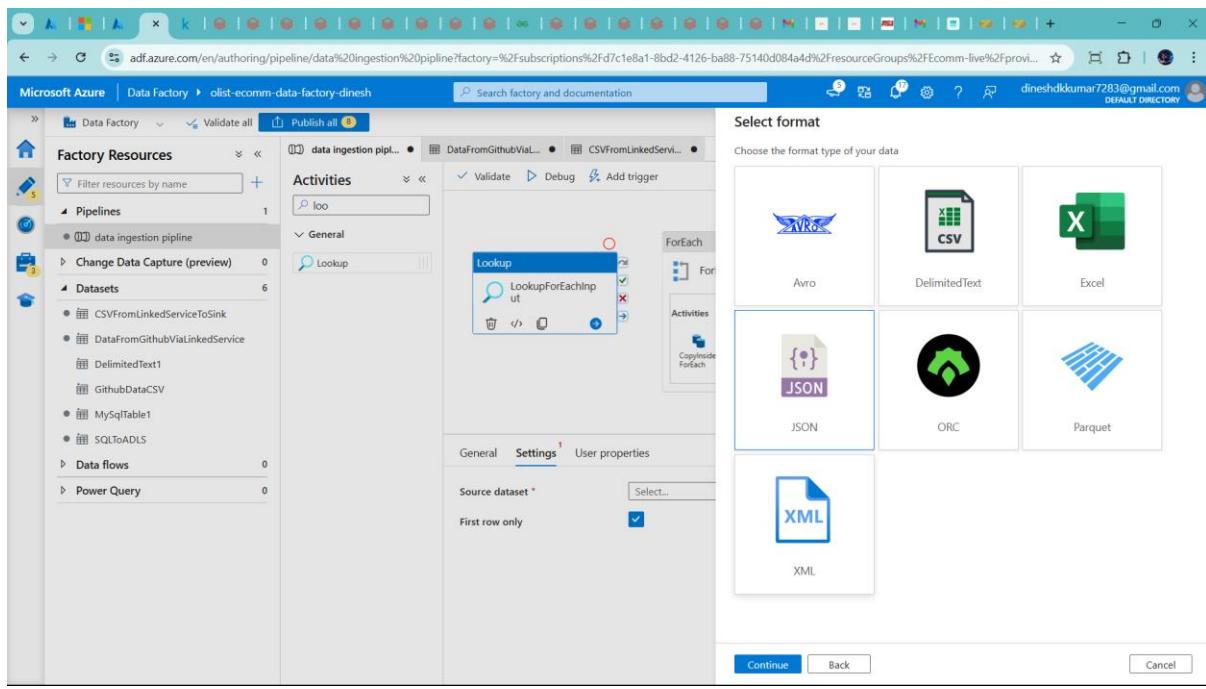


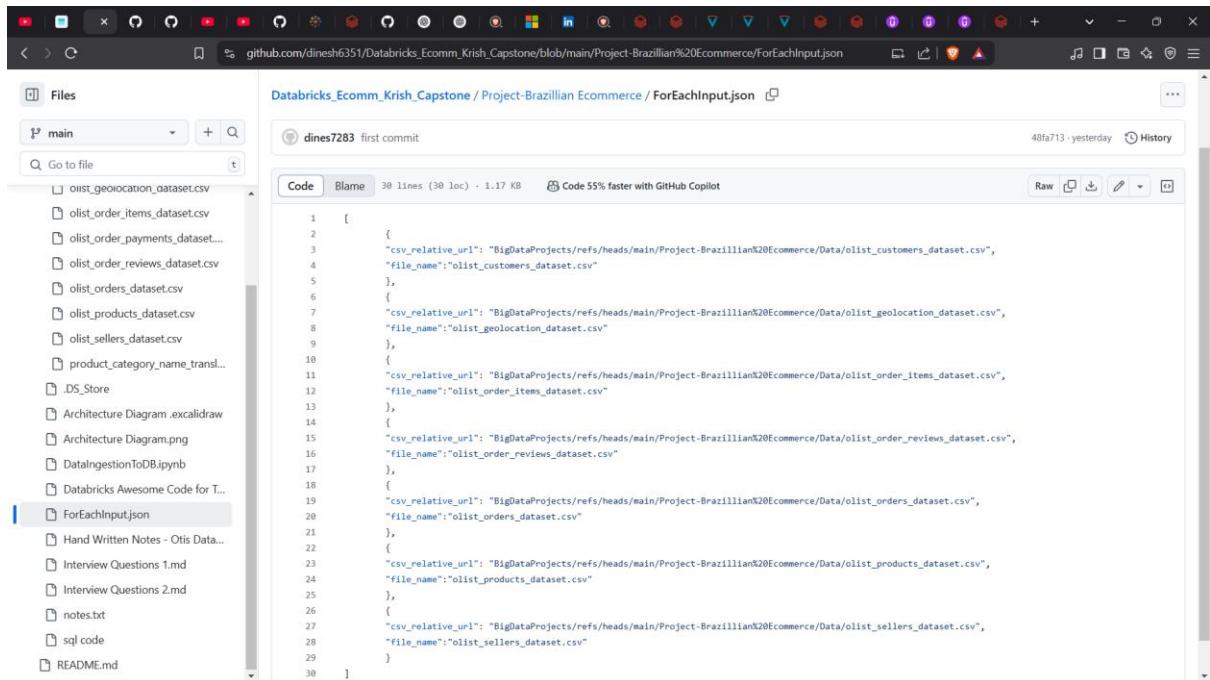




→Add Lookup





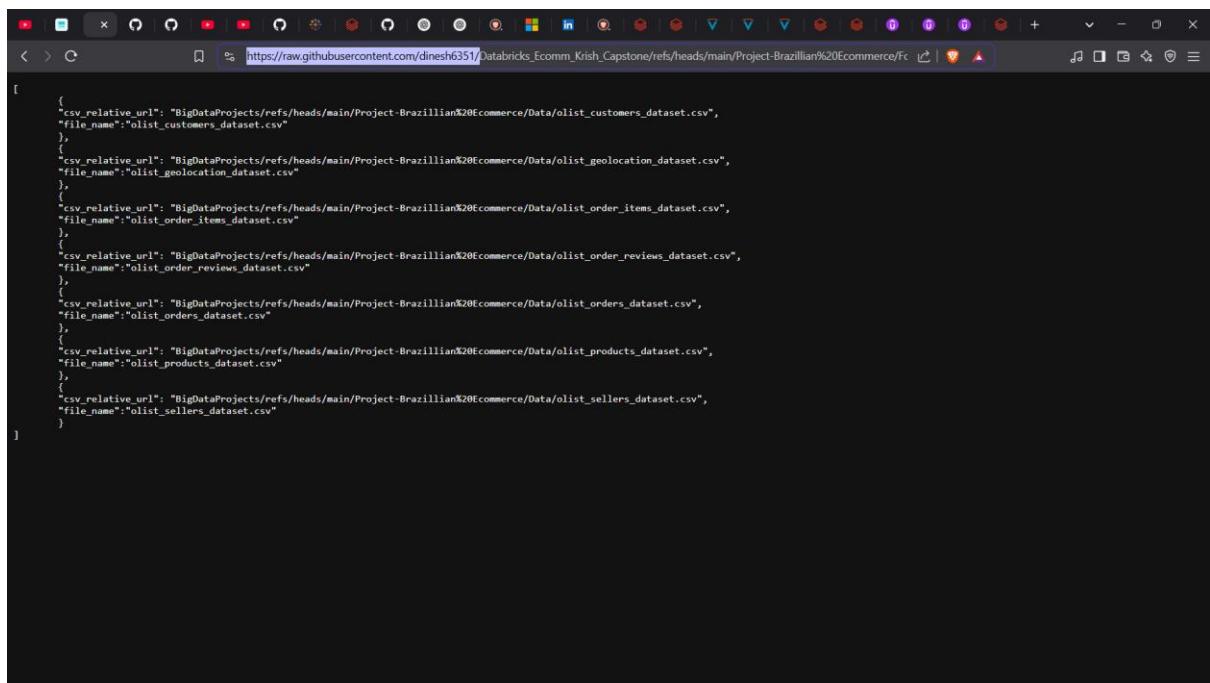


Databricks_Ecomm_Krish_Capstone / Project-Brazilian Ecommerce / ForEachInput.json

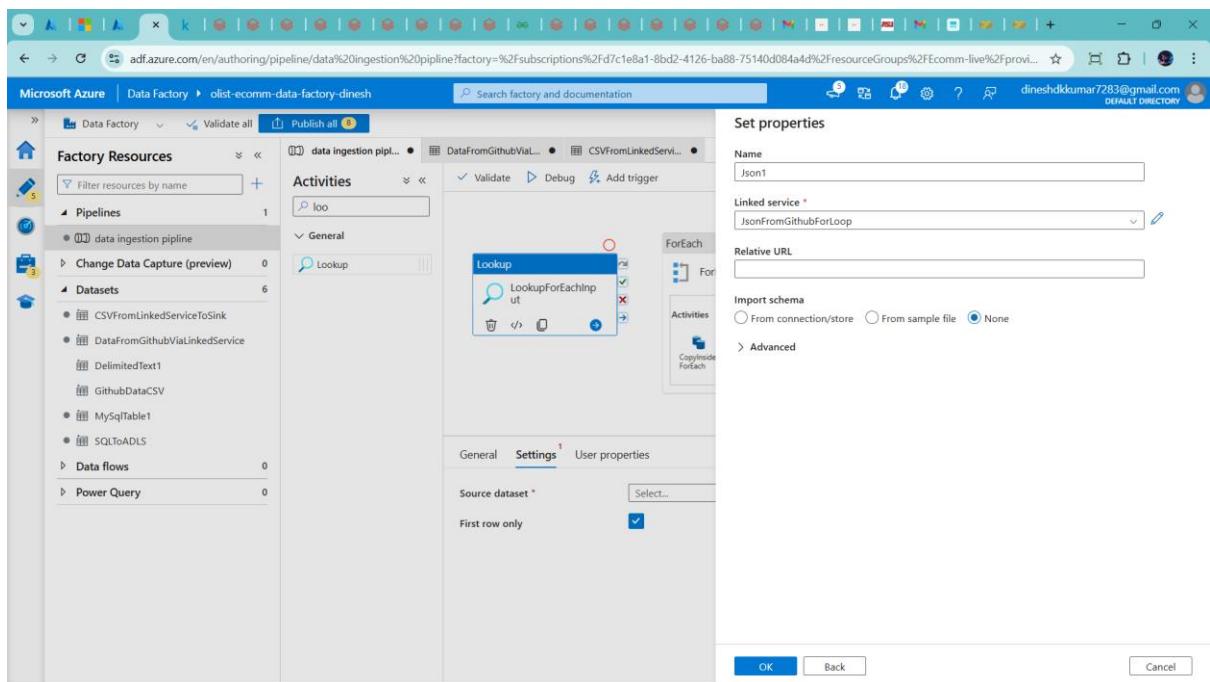
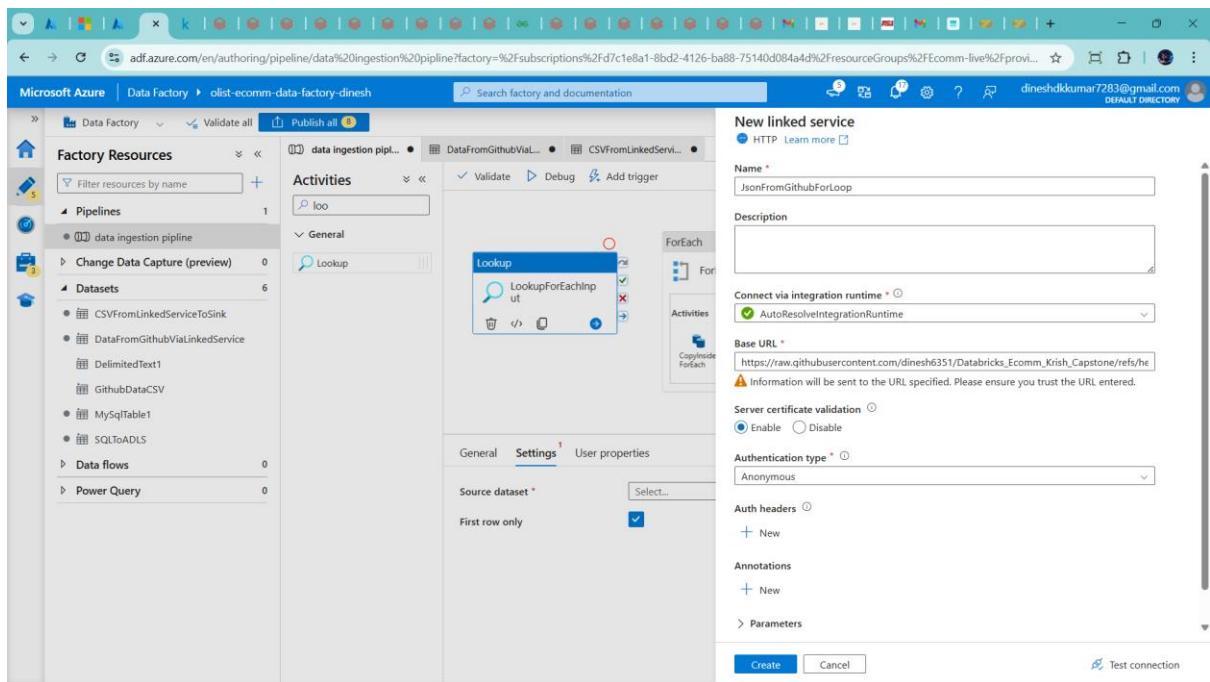
dinesh7283 first commit · 48fa713 · yesterday · History

Code Blame 30 lines (30 loc) · 1.17 KB · Code 55% faster with GitHub Copilot

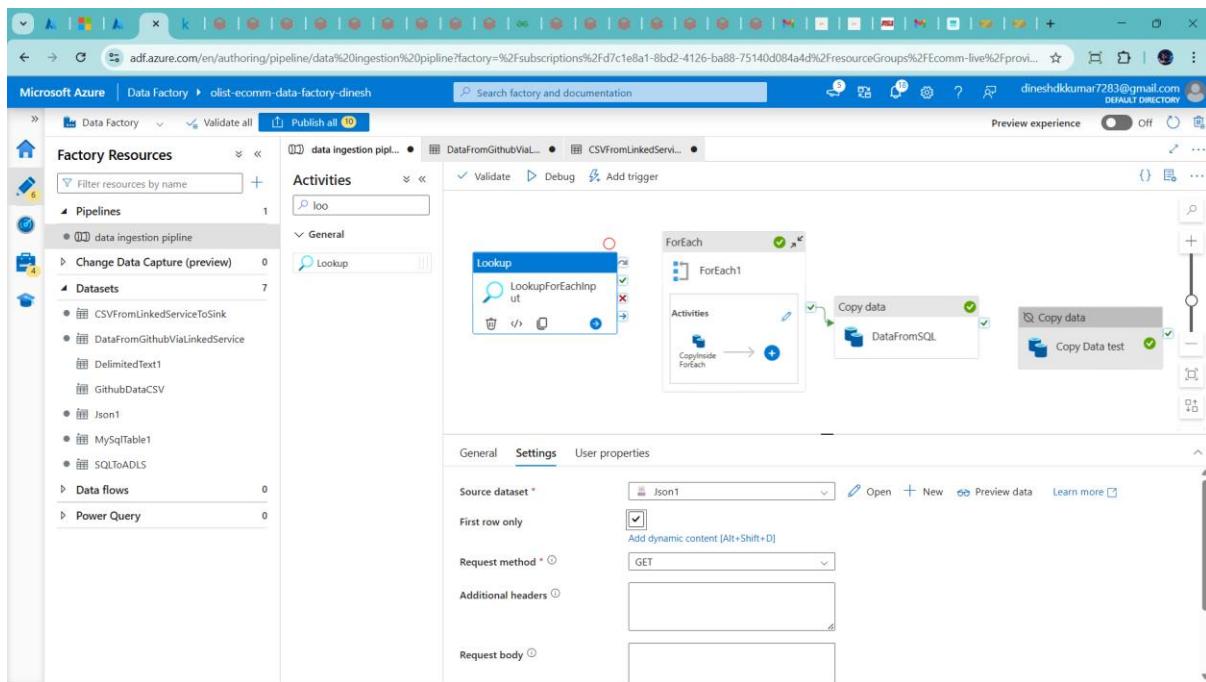
```
[{"csv_relative_url": "BigDataProjects/ref/heads/main/Project-Brazilian%20Ecommerce/Data/olist_customers_dataset.csv", "file_name": "olist_customers_dataset.csv"}, {"csv_relative_url": "BigDataProjects/ref/heads/main/Project-Brazilian%20Ecommerce/Data/olist_geolocation_dataset.csv", "file_name":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



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

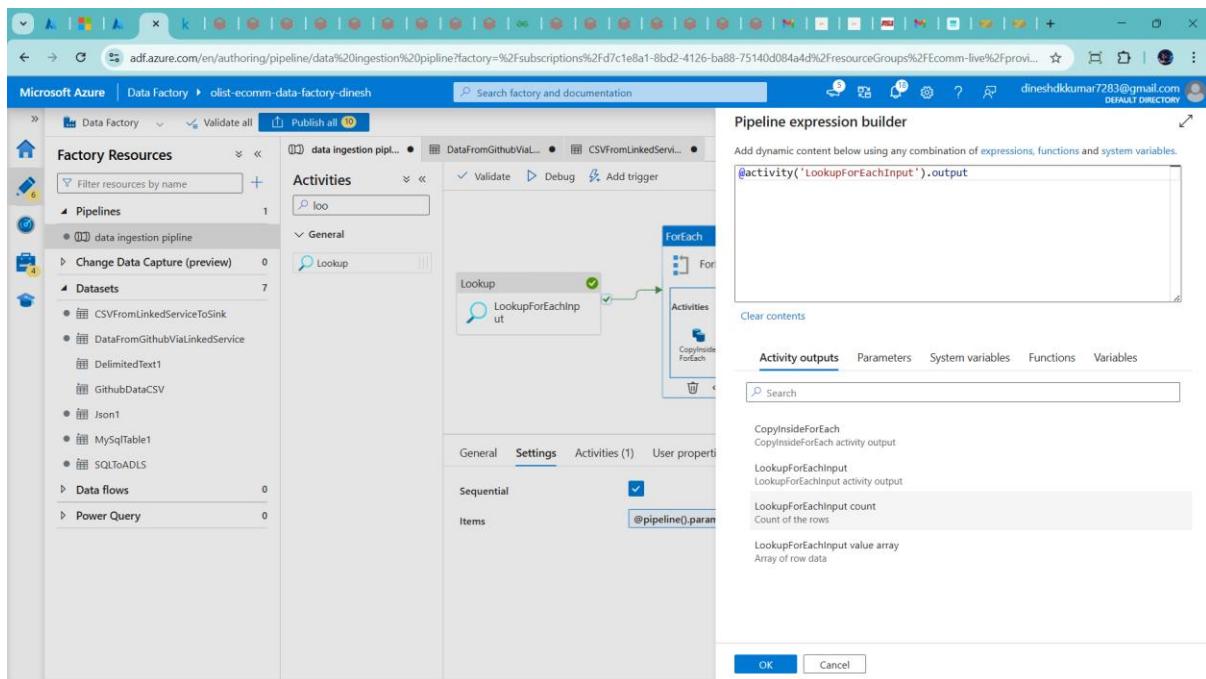


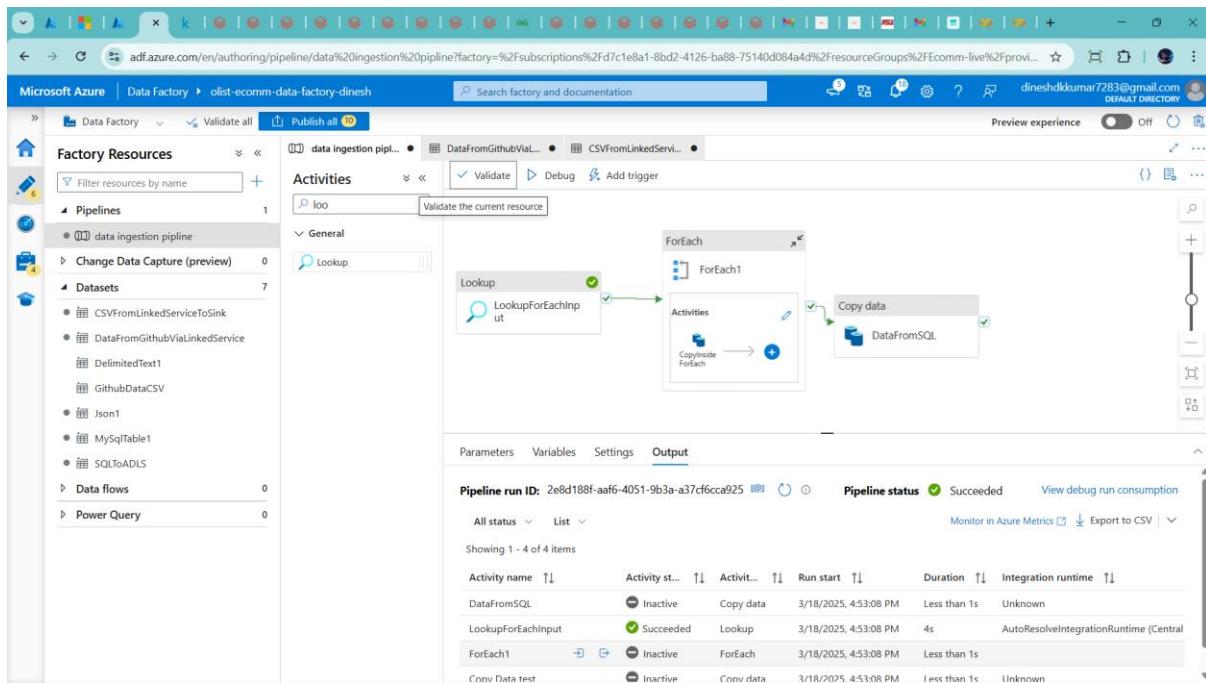
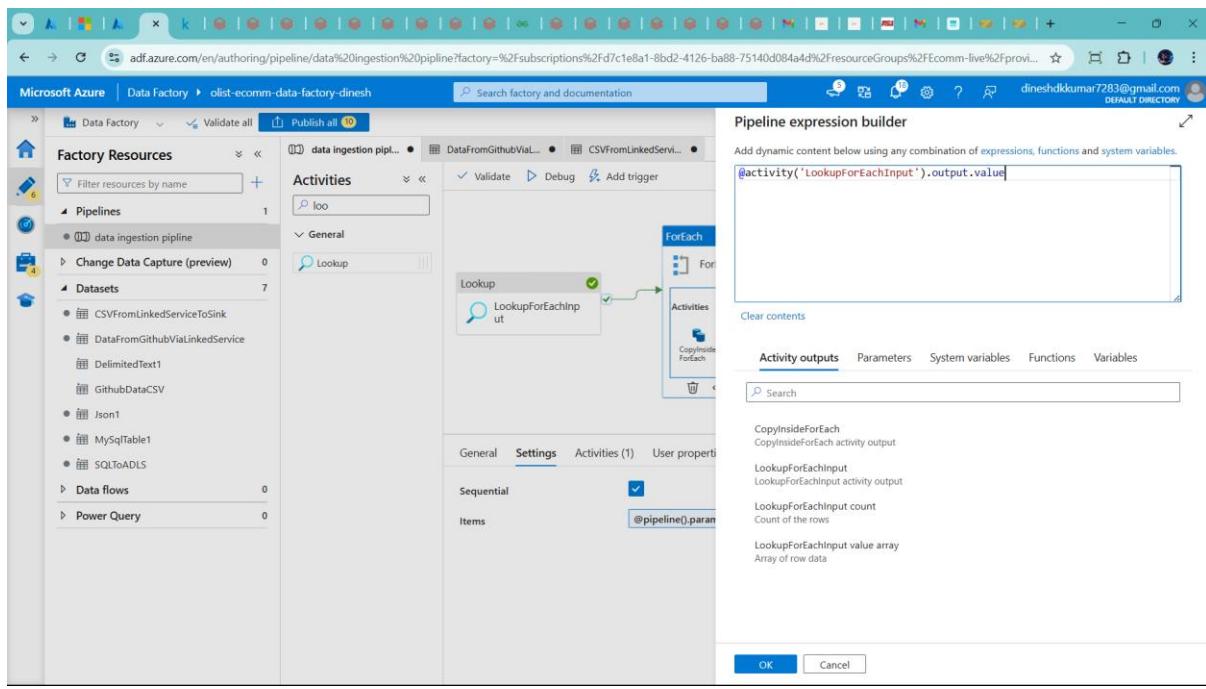
→ uncheck first row only



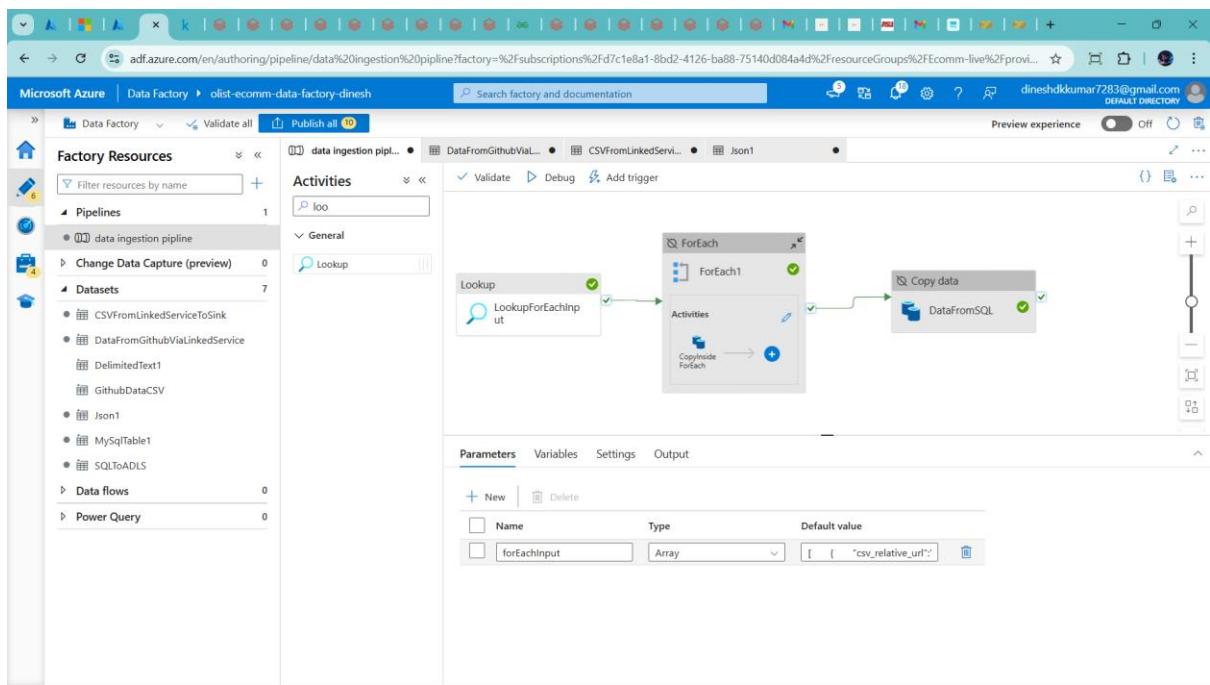
Change foreach as well

→ click setting

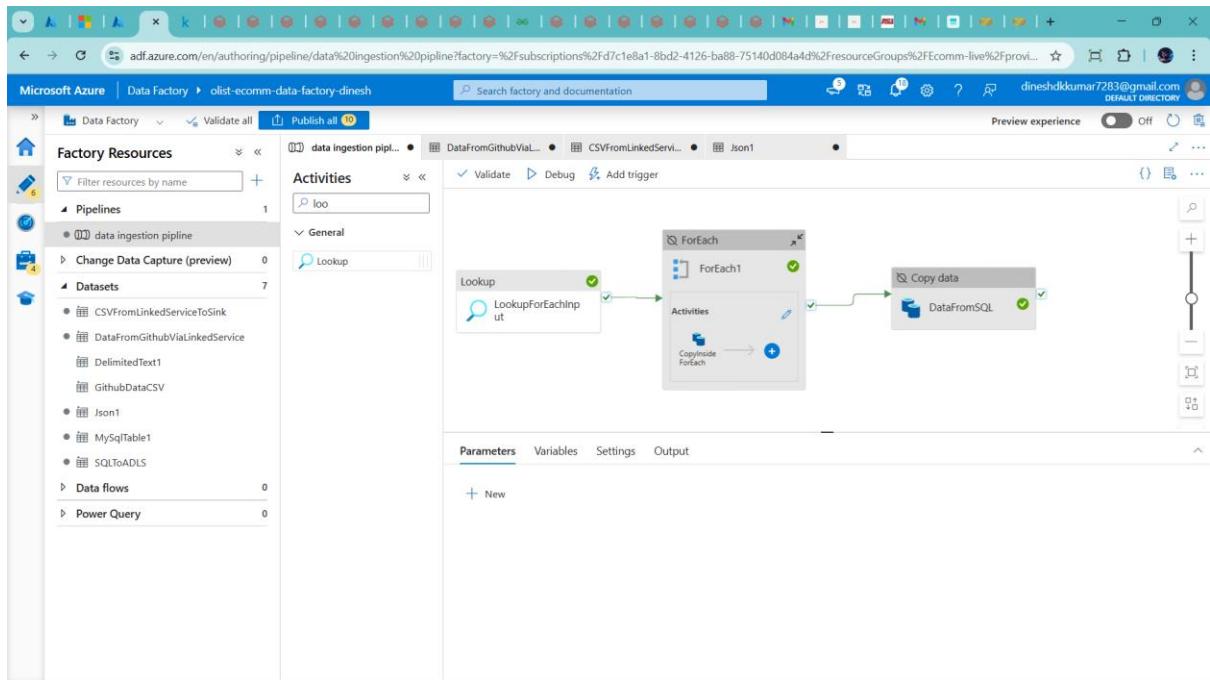




→Now Debug



→ Remove



The screenshot shows the Microsoft Azure Storage Container Overview page for the 'olistdata' container. The page header includes the URL 'portal.azure.com/#view/Microsoft_Azure_Storage/ContainerMenuBlade/~/overview/storageAccountId/%2Fsubscriptions%2Fd7c1e8a1-8bd2-4126-ba88-75140d084a4d%2Fresource...', the user 'dineshkumar7283@...', and the 'DEFAULT DIRECTORY' indicator. The main content area has tabs for 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', and 'Settings'. The 'Overview' tab is selected. It displays a table of blobs with columns: Name, Modified, Access tier, Archive status, Blob type, Size, and Lease state. The blobs listed are:

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[-]						
olist_customers_dataset.csv	3/18/2025, 5:40:41 PM	Hot (Inferred)		Block blob	8.62 MiB	Available
olist_geolocation_dataset.csv	3/18/2025, 5:40:58 PM	Hot (Inferred)		Block blob	58.44 MiB	Available
olist_order_items_dataset.csv	3/18/2025, 5:41:13 PM	Hot (Inferred)		Block blob	14.72 MiB	Available
olist_order_reviews_dataset.csv	3/18/2025, 5:41:27 PM	Hot (Inferred)		Block blob	13.68 MiB	Available
olist_orders_dataset.csv	3/18/2025, 5:42:53 PM	Hot (Inferred)		Block blob	16.84 MiB	Available
olist_products_dataset.csv	3/18/2025, 5:43:07 PM	Hot (Inferred)		Block blob	2.27 MiB	Available
olist_sellers_dataset.csv	3/18/2025, 5:43:21 PM	Hot (Inferred)		Block blob	170.61 KiB	Available

Data Ingestion

Used Azure Data factory with http & a SQL server.

- parametrization
- for each activity
- lookup

The screenshot shows the Microsoft Azure Marketplace search results for 'databricks'. The search bar at the top contains 'databricks'. Below it, there are filters: Pricing : All, Operating System : All, Publisher Type : All, Product Type : All, and Publisher name : All. A message says 'New! Get AI-generated suggestions for 'databricks'' with a 'View suggestions' button. The results section shows 1 to 20 of 69 results. Each result card includes a thumbnail, the product name, the publisher, a brief description, price (e.g., 'Price varies'), and a 'Create' button. The cards for 'Azure Databricks' and 'Unravel for Azure Databricks' are highlighted.

→ I choose premium

The screenshot shows the 'Create an Azure Databricks workspace' wizard. At the top, a green banner says 'Validation Succeeded'. Below it, tabs include Basics, Networking, Encryption, Security & compliance, Tags, and Review + create (which is underlined). The Basics tab shows the following configuration:

Workspace name	olist-spark-workspace
Subscription	Azure for Students
Resource group	Ecomm-live
Region	Central India
Pricing Tier	premium
Managed Resource Group name	ecomm-databricks-resource-group

The Networking tab shows:

Deploy Azure Databricks workspace with Secure Cluster Connectivity (No Public IP)	No
Deploy Azure Databricks workspace in your own Virtual Network (VNet)	No

The Encryption tab shows:

Enable Infrastructure Encryption	No
----------------------------------	----

At the bottom are buttons for Create, < Previous, and Download a template for automation.

Azure Databricks

Azure Databricks



- Spark powered
- Integrated with Azure
- Handles big Data easily
- Great for machine learning

Home > Ecomm-live_olist-spark-workspace | Overview >

olist-spark-workspace Azure Databricks Service

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Settings

Monitoring

Automation

Help

Status : Active

Resource group : Ecomm-live

Location : Central India

Subscription : Azure for Students

Subscription ID : d7c1e8a1-8bd2-4126-ba88-75140d084a4d

Tags (edit) : Add tags

Managed Resource Group : ecomm-databricks-resource-group

URL : <https://adb-3220087795989142.2.azuredatabricks.net>

Pricing Tier : Premium (+ Role-based access controls) (Click to change)

Launch Workspace

Documentation

Getting Started

Import Data from File

Import Data from Azure Storage

Notebook

Admin Guide

Link Azure ML workspace

Azure Databricks Workflow

1. Read data from ADLS gen2
2. Perform basic transformation like cleaning, renaming & filtering
3. Use join operation to integrate multiple dataset.
4. Enrich the data via MongoDB.
5. Perform aggregation & derive insights
6. Write final data back to ADLS gen2.

→ Launch Databricks workspace

The screenshot shows the Microsoft Azure portal with the URL <https://portal.azure.com/#@dineshkumar7283@gmail.onmicrosoft.com/resource/subscriptions/d7c1e8a1-8bd2-4126-ba88-75140d084a4d/resourceGroups/Ecomm-live/providers/Microsoft/AzureDatabricks/workspaces/olist-spark-workspace>. The page displays the 'olislist-spark-workspace' Azure Databricks Service workspace. The 'Overview' tab is selected, showing details like Status: Active, Resource group: Ecomm-live, Location: Central India, Subscription: Azure for Students, and Subscription ID: d7c1e8a1-8bd2-4126-ba88-75140d084a4d. A red icon representing three stacked rectangles is prominently displayed in the center. Below it is a blue 'Launch Workspace' button. To the right of the main content area is a vertical JSON View panel.

→ Create compute → it is like hardware

The screenshot shows the Databricks Compute page at the URL <https://adb-3220087795989142.2.azuredatabricks.net/compute?o=3220087795989142>. The left sidebar shows navigation options such as New, Workspace, Recents, Catalog, Workflows, Compute (which is selected), Marketplace, SQL, and Data Engineering. The main content area is titled 'Compute' and shows a table with columns: State, Name, Policy, Runtime, Active mem..., Active cores, Active DBU ..., Source, Creator, Notebooks, and a gear icon. A large red '+' icon is centered above the table. Below the table, a message reads 'No compute' and 'Create compute to run workloads from your notebooks and jobs. Learn more about best practices for compute configuration'. A blue 'Create compute' button is located at the bottom of this section.

Dinesh km's Cluster

Policy: Unrestricted

Access mode: Single user access

Performance: Runtime: 15.4 LTS (Scala 2.12, Spark 3.5.0), Use Photon Acceleration checked

Node type: Standard_D4ds_v5

Tags: Add tags

Summary: 1 Driver, 16 GB Memory, 4 Cores, Runtime: 15.4 x-scala2.12, Unity Catalog: Photon, Standard_D4ds_v5: 2 DBU/h

→ click compute

Compute

All-purpose compute

State	Name	Policy	Runtime	Active mem...	Active cores	Active DBU ...	Source	Creator	Notebooks	⋮
Running	Dinesh km's Cluster	-	15.4	-	-	-	UI	Dinesh km	-	⋮

→ Create to add data in MongoDB

→ Go to file.io

The screenshot shows the file.io MongoDB dashboard for the database 'olistDataNoSQL'. The interface includes a sidebar with user information (dinesh km, dineshkumar63519, 500@gmail.com) and navigation links for Databases, Standard Tier, Billing, API, and Status. The main area displays a green shield icon, connection details (Host: 9o6fs.h.filess.io, Database: olistDataNoSQL_poolca, User: olistDataNoSQL_poolca, Port: 27018), and a password field. It also shows a 'Web Client' button, a progress bar for size (0.00 MB / 10 MB), and a 'Request Backup' button. A 'Code' button is visible at the bottom right.

The screenshot shows the file.io MongoDB dashboard with the 'Generate code' section active. The user has selected 'Python' from a dropdown menu. The generated code is as follows:

```
1 # importing module
2 from pymongo import MongoClient
3
4 hostname = "9o6fs.h.filess.io"
5 database = "olistDataNoSQL_poolcanpie"
6 port = "27018"
7 username = "olistDataNoSQL_poolcanpie"
8 password = "fa9f830f3ca3365abdeaf4a10a4041c522fc3"
9
10 uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database
11
12 # Connect with the portnumber and host
13 client = MongoClient(uri)
14
15 # Access database
16 mydatabase = client[database]
```

If you want to contribute to this project and add support for your language, you can create a pull request in our [GitHub repository](#).

→ Copy the code and add in googleColab

The screenshot shows a Jupyter Notebook interface with a single code cell containing Python code. The code attempts to import MongoClient from pymongo and connect to a MongoDB database. However, it fails because the module has not been installed. The error message 'ModuleNotFoundError' is displayed, along with the stack trace showing the import path.

```
Inserted records 22501 to 23000 successfully.  
Inserted records 23001 to 23500 successfully.  
Inserted records 23501 to 24000 successfully.  
Inserted records 24001 to 24500 successfully.  
Inserted records 24501 to 25000 successfully.  
Inserted records 25001 to 25500 successfully.  
Inserted records 25501 to 26000 successfully.  
Inserted records 26001 to 26500 successfully.  
  
[x] # importing module  
from pymongo import MongoClient  
  
hostname = "906fs.h.filess.io"  
database = "olistDataNoSQL_poolcanpie"  
port = "2018"  
username = "olistDataNoSQL_poolcanpie"  
password = "faaf830f3ca3365abdeafafda10a041c522fc3"  
  
uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database  
  
# Connect with the portnumber and host  
client = MongoClient(uri)  
  
# Access database  
mydatabase = client[database]  
  
ModuleNotFoundError: No module named 'pymongo'  Traceback (most recent call last)  
c:\python\input_1-146fb2114ce> in <cell line: 6>()  
      1 # importing module  
----> 2 from pymongo import MongoClient  
      3  
      4 hostname = "906fs.h.filess.io"  
      5 database = "olistDataNoSQL_poolcanpie"
```

The screenshot shows the same Jupyter Notebook after the user has run the command 'pip install pymongo'. The output indicates that the package was successfully downloaded and installed. The code cell below then runs without errors, demonstrating successful connection to the MongoDB database.

```
[2] pip install pymongo  
Collecting pymongo  
  Downloading pymongo-4.11.3-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (22 kB)  
Collecting dsspython<0.0.6,>=1.16.0 (from pymongo)  
  Downloading dsspython-0.0.6-py3-none-any.whl.metadata (5.8 kB)  
Collecting dsspython<2.7.0,>=2.7.0 (from pymongo)  
  Downloading dsspython-2.7.0-py3-none-any.whl (313 kB)  
    1.8/1.4 MB 28.1 MB/s eta 0:00:00  
  Downloading dsspython-2.7.0-py3-none-any.whl (313 kB)  
    313.6/313.6 KB 19.5 MB/s eta 0:00:00  
Installing collected packages: dsspython, pymongo  
Successfully installed dsspython-2.7.0 pymongo-4.11.3  
  
[x] # importing module  
from pymongo import MongoClient  
  
hostname = "906fs.h.filess.io"  
database = "olistDataNoSQL_poolcanpie"  
port = "2018"  
username = "olistDataNoSQL_poolcanpie"  
password = "faaf830f3ca3365abdeafafda10a041c522fc3"  
  
uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database  
  
# Connect with the portnumber and host  
client = MongoClient(uri)  
  
# Access database  
mydatabase = client[database]
```

→add file data →go to github

Name	Last commit message	Last commit date
..		2 days ago
olist_customers_dataset.csv	first commit	2 days ago
olist_geolocation_dataset.csv	first commit	2 days ago
olist_order_items_dataset.csv	first commit	2 days ago
olist_order_payments_dataset.csv	first commit	2 days ago
olist_order_reviews_dataset.csv	first commit	2 days ago
olist_orders_dataset.csv	first commit	2 days ago
olist_products_dataset.csv	first commit	2 days ago
olist_sellers_dataset.csv	first commit	2 days ago
product_category_name_translation.csv	first commit	2 days ago

github.com/dinesh6351/.../product_category_name_translation.csv

→ download it

1	product_category_name	product_category_name_english
2	beleza_saude	health_beauty
3	informatica_acessorios	computers_accessories
4	automotivo	auto
5	cama_banho	bed_bath_table
6	moveis_decoracao	furniture_decor
7	esporte_lazer	sports_leisure
8	perfumaria	perfumery
9	utilidades_domesticas	housewares
10	telefonia	telephony
11	relogios_presentes	watches_gifts
12	alimentos_bebidas	food_drink
13	bebés	baby

→ add in google colab

File IntentionToSQL.ipynb

```
# ACCESS TO DATABASE
mydatabase = client[database]

# prompt: read the /content/product_category_name_translation.csv and create a collection and upload mongodb
import pandas as pd
from pymongo import MongoClient

# MongoDB connection details (replace with your own)
hostname = "900fs.h.fileless.io"
database = "olistDataNoSQL_poolcanpie"
port = "27018"
username = "olistDataNoSQL_poolcanpie"
password = "fa9f830f3ca3365abdea1af4a10a4041c522fc3"

uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database

# Connect to MongoDB
client = MongoClient(uri)
db = client[database]

# CSV file path
csv_file_path = "/content/product_category_name_translation.csv"

# Collection name
collection_name = "product_category_name_translation"

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(csv_file_path)
```

File IntentionToSQL.ipynb

```
password = "fa9f830f3ca3365abdea1af4a10a4041c522fc3"

uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database

# Connect to MongoDB
client = MongoClient(uri)
db = client[database]

# CSV file path
csv_file_path = "/content/product_category_name_translation.csv"

# Collection name
collection_name = "product_category_name_translation"

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(csv_file_path)

# Convert DataFrame to a list of dictionaries
data = df.to_dict('records')

# Access the collection
collection = db[collection_name]

# Insert the data into the collection
collection.insert_many(data)

print(f'Data uploaded to MongoDB collection "{collection_name}" successfully!')

client.close()
```

The screenshot shows a Google Colab notebook titled "DataIntentionToSQL.ipynb". The code cell contains Python code for connecting to a MongoDB database and inserting data from a CSV file. The output of the cell shows a success message: "Data uploaded to MongoDB collection 'product_category_name_translation' successfully!". The interface includes a sidebar with file navigation, a status bar showing disk usage (70.81 GB available), and a bottom bar with a progress indicator.

```
uri = "mongodb://" + username + ":" + password + "@" + hostname + ";" + port + "/" + database
client = MongoClient(uri)
db = client[database]

# csv file path
csv_file_path = "/content/product_category_name_translation.csv"

# collection name
collection_name = "product_category_name_translation"

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(csv_file_path)

# Convert DataFrame to a list of dictionaries
data = df.to_dict('records')

# Access the collection
collection = db[collection_name]

# Insert the data into the collection
collection.insert_many(data)

print(f'Data uploaded to MongoDB collection "{collection_name}" successfully!')

client.close()
```

The screenshot shows the filess.io MongoDB interface. On the left, the "CONNECTIONS" sidebar lists a connection named "olistDataNoSQL_poolcanpie - unsaved". The main area, titled "COLLECTIONS", shows a single collection named "test" with 0 rows. The bottom status bar indicates the connection is "Connected" to "MongoDB 7.0.2" at "2 minutes ago".

The screenshot shows a MongoDB browser window with the following details:

- CONNECTIONS:** olistDataNoSQL_poolcanpie - unsaved
- COLLECTIONS:** Collections (2) - product_category_name_translation
- FILTERS:** test: 0 rows
- MACROS:**

_id	product_category_name	product_category_name_english
ObjectID("67da5524c9281859c7897f")	beleza_saude	health_beauty
ObjectID("67da5524c9281859c7897f")	informatica_acessorios	computers_acessorios
ObjectID("67da5524c9281859c7897f")	automotivo	auto
ObjectID("67da5524c9281859c7897f")	cama_mesa_banho	bed_bath_table
ObjectID("67da5524c9281859c7897f")	moveis_decoracao	furniture_decor
ObjectID("67da5524c9281859c7897f")	esporte_lazer	sports_leisure
ObjectID("67da5524c9281859c7897f")	perfumaria	perfumery
ObjectID("67da5524c9281859c7897f")	utilidades_domesticas	housewares
ObjectID("67da5524c9281859c7897f")	telefonia	telephony
ObjectID("67da5524c9281859c7897f")	religios_presentes	watches_gifts
ObjectID("67da5524c9281859c7897f")	alimentos_bebidas	food_drink
ObjectID("67da5524c9281859c7897f")	bebidas	baby
ObjectID("67da5524c9281859c7897f")	papelaria	stationery
ObjectID("67da5524c9281859c7897f")	tablets_impressao_imagem	tablets_printing_image
ObjectID("67da5524c9281859c7897f")	brinquedos	toys
ObjectID("67da5524c9281859c7897f")	telefonia_fixa	fixed_telephony
ObjectID("67da5524c9281859c7897f")	ferramentas_jardim	garden_tools
ObjectID("67da5524c9281859c7897f")	fashion_bolsas_e_acessorios	fashion_bags_accessories
ObjectID("67da5524c9281859c7897f")	eletroportateis	small_appliances
ObjectID("67da5524c9281859c7897f")	consoles_games	consoles_games
ObjectID("67da5524c9281859c7897f")	audios	audio
ObjectID("67da5524c9281859c7897f")	fashion_calcados	fashion_shoes
ObjectID("67da5524c9281859c7897f")	cool_stuff	cool_stuff
ObjectID("67da5524c9281859c7897f")	malas_acessorios	luggage_accessories
ObjectID("67da5524c9281859c7897f")	climatizacao	air_conditioning

→connect ADLS gen 2 to databricks

The screenshot shows a Microsoft Learn article with the following content:

Step 7: Connect to Azure Data Lake Storage using python

These properties are available from the *Settings > Properties tab of an Azure Key Vault in your Azure portal.

Click the Create button.

1. Navigate to your Azure Databricks workspace and create a new python notebook.

2. Run the following python code, with the replacements below, to connect to Azure Data Lake Storage.

```
Python
service_credential = dbutils.secrets.get(scope="<scope>",key="<service-credential-key>")
spark.conf.set("fs.azure.account.auth.type.<storage-account>.dfs.core.windows.net", "service_principal")
spark.conf.set("fs.azure.account.oauth.provider.type.<storage-account>.dfs.core.windows.net", "OAuth2ClientAuthProvider")
spark.conf.set("fs.azure.account.oauth2.client.id.<storage-account>.dfs.core.windows.net", "<client-id>")
spark.conf.set("fs.azure.account.oauth2.client.secret.<storage-account>.dfs.core.windows.net", "<client-secret>")
spark.conf.set("fs.azure.account.oauth2.client.endpoint.<storage-account>.dfs.core.windows.net", "<endpoint>")
```

Replace

- **<scope>** with the secret scope name from step 5.
- **<service-credential-key>** with the name of the key containing the client secret.

Training

Large-Scale Data Processing with Azure Data Lake Storage Gen2 - Training

Large-Scale Data Processing with Azure Data Lake Storage Gen2

Certification

Microsoft Certified: Azure Data Engineer Associate - Certifications

Demonstrate understanding of common data engineering tasks to implement and manage data engineering workloads on Microsoft Azure, using a...

Documentation

Connect to Azure Data Lake Storage and Blob Storage - Azure Databricks

Learn how to configure Azure Databricks to use the ABFS driver to read and write data stored on Azure Data Lake Storage and Blob Storage.

Mounting cloud object storage on Azure Databricks - Azure Databricks

Learn how to view, create, and manage tables and databases in Azure Databricks.

Tutorial: Azure Data Lake Storage, Azure Databricks & Spark - Azure Storage

This tutorial shows how to run Spark queries on an Azure Databricks cluster to access data in an Azure Data Lake Storage storage account.

Show 4 more

→copy and add in dataricks

```
spark.conf.set("fs.azure.account.auth.type.<storage-account>.dfs.core.windows.net", "OAuth")
spark.conf.set("fs.azure.account.oauth.provider.type.<storage-account>.dfs.core.windows.net", "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider")
spark.conf.set("fs.azure.account.oauth2.client.id.<storage-account>.dfs.core.windows.net", "<application-id>")
spark.conf.set("fs.azure.account.oauth2.client.secret.<storage-account>.dfs.core.windows.net", "service_credential")
spark.conf.set("fs.azure.account.oauth2.client.endpoint.<storage-account>.dfs.core.windows.net", "https://login.microsoftonline.com/<directory-id>/oauth2/token")
```

```
storage_account = "<storage-account>"
application_id = "<application-id>"
directory_id = "<directory-id>"
service_credential = "<service-credential>"

spark.conf.set("fs.azure.account.auth.type.(storage_account).dfs.core.windows.net", "OAuth")
spark.conf.set("fs.azure.account.oauth.provider.type.(storage_account).dfs.core.windows.net", "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider")
spark.conf.set("fs.azure.account.oauth2.client.id.(storage_account).dfs.core.windows.net", application_id)
spark.conf.set("fs.azure.account.oauth2.client.secret.(storage_account).dfs.core.windows.net", service_credential)
spark.conf.set("fs.azure.account.oauth2.client.endpoint.(storage_account).dfs.core.windows.net", "https://login.microsoftonline.com/<directory_id>/oauth2/token")
```

-->change bases on Ai

→search app registration in azure

Microsoft Azure

Ecomm-live

Home >

Overview

Search

Services (99+)

Marketplace (30)

All

App registrations

App Services

Azure Cosmos DB

Azure Database for MySQL servers

Essentials

Subscription (move) : Azure for...

Subscription ID : d7c1e8a1

Tags (edit) : Add tags

Resources

Recommendations

Filter for any field...

Showing 1 to 3 of 3 records

Name

olist-ecomm-data-fac

olist-spark-workspace

olistdatastoragedlines

Documentation

Create your first function in the Azure portal

Quickstart: Create a Node.js web app - Azure App Service

Create Standard workflows with Visual Studio Code - Azure Logic Apps

Create an App Service app using a Terraform template - Azure App Service

Searching all subscriptions.

Assign tags

Move

Delete

Export template

JSON View

3 Succeeded

Central India

No grouping

List view

Location ↑↓

Central India

Central India

Central India

< Previous Page 1 of 1 Next >

Give feedback

→ Click new registration

Microsoft Azure

Home >

App registrations

Search resources, services, and docs (G+)

Copilot

dineshkumar7283@q... DEFAULT DIRECTORY

+ New registration

Endpoints

Troubleshoot

Refresh

Download

Preview features

Got feedback?

Owned applications

All applications

Deleted applications

Applications from personal account

Start typing a display name or application (client) ID to filter these results...

Add filters

This account isn't listed as an owner of any applications in this directory.

View all applications in the directory

View all applications from personal account

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

Microsoft Azure

Home > App registrations >

Register an application

...
iName
The user-facing display name for this application (this can be changed later).
olist-app-registration-db-adfs

Supported account types
Who can use this application or access this API?
 Accounts in this organizational directory only (Default Directory only - Single tenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
 Personal Microsoft accounts only
Help me choose...

Redirect URI (optional)
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.
Select a platform e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from Enterprise applications.

By proceeding, you agree to the Microsoft Platform Policies [Learn more](#)

Register

→fill and click registration

Microsoft Azure

Home > App registrations > olist-app-registration-db-adfs

olist-app-registration-db-adfs

Search Delete Endpoints Preview features

Overview

Essentials

Display name	:	olist-app-registration-db-adfs
Application (client) ID	:	260e6732-04d0-4d7c-be5d-fba04b30aa93
Object ID	:	b3fc51cd-9c58-4b34-a650-f5dd8f0020c3
Directory (tenant) ID	:	1eadda0c-f714-4f5e-8af6-f435b63d1d8e
Supported account types	:	My organization only

Client credentials : Add a certificate or secret
Redirect URLs : Add a Redirect URI
Application ID URI : Add an Application ID URI
Managed application in L... : olist-app-registration-db-adfs

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

Get Started Documentation

Build your application with the Microsoft identity platform

The Microsoft identity platform is an authentication service, open-source libraries, and application management tools. You can create modern, standards-based authentication solutions, access and protect APIs, and add sign-in for your users and customers. [Learn more](#)

→click manage→certificate and secrets →click client secrets

Home > App registrations > olist-app-registration-db-adfs

olist-app-registration-db-adfs | Certificates & secrets

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

Certificates (0) Client secrets (0) 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

No client secrets have been created for this application.

→click add

Home > App registrations > olist-app-registration-db-adfs

olist-app-registration-db-adfs | Certificates & secrets

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

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

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

+ New client secret

Description	Expires	Value	Secret ID

No client secrets have been created for this application.

Add a client secret

Description: db-client-secret

Expires: Recommended: 180 days (6 months)

Add Cancel

→value will be use →so copy the value

olist-app-registration-db-adfs | Certificates & secrets

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

Description Expires Value Secret ID

db-client-secret 9/15/2025 8BYBQ-mOhtnERvXaeIJDt-JHAIkTCPlixE... da2ddc55-8739-4656-b5ea-294bc8f0af0c

→ now add application in this databricks

Display name : olist-app-registration-db-adfs

Application (client) ID : 260e6732-04d0-4d7c-be5d-fba04b30aa93

Object ID : b3fc51cd-9c58-4b34-aa60-f5dd8f0020c3

Directory (tenant) ID : 1eadda0c-f714-4f5e-8af6-f435b63d1dbe

Supported account types : My organization only

Client credentials : 0_certificate_1.secret

Redirect URLs : Add a Redirect URI

Application ID URI : Add an Application ID URI

Managed application in ... : olist-app-registration-db-adfs

→ copy as scrted

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is `portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationMenuBlade/~/Credentials/appId/260e6732-04d0-4d7c-be5d-fba04b30aa93/objectId/b3fc51cd-9c58-4b34-aa60-...`. The user is signed in as `dineshkumar7283@gmail.com`.

The main content area displays the 'Certificates & secrets' blade for the application 'olist-app-registration-db-adfs'. The 'Client secrets (1)' tab is selected. A single client secret named 'db-client-secret' is listed, showing its description, expiration date (9/15/2025), value (copied to clipboard), and secret ID (da2ddc55-8739-4656-b5ea-294bc8f0af0c).

→copy name as storage account

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is `portal.azure.com/#@dineshkumar7283@gmail.onmicrosoft.com/resource/subscriptions/d7c1e8a1-8bd2-4126-ba88-75140d084a4d/resourceGroups/Ecomm-live/providers/Microsoft...`. The user is signed in as `dineshkumar7283@gmail.com`.

The main content area displays the 'Storage account' blade for the storage account 'olistdatastoragedinesh'. The 'Overview' section provides details such as Resource group (Ecomm-live), Location (centralindia), Subscription (Azure for Students), Subscription ID (d7c1e8a1-8bd2-4126-ba88-75140d084a4d), Disk state (Available), and Created (3/18/2025, 3:00:13 PM). The 'Properties' tab is also visible.

```

storage_account = "olistdatastoragedinedesh"
application_id = "260e6732-04d8-4d7c-be5d-fba04b30aa93"
directory_id = "1eaddad0c-f714-4f5e-8af6-f435b63d1d8e"
service_credential = "BBY8Q=m0htnErVXaei3Dt-JHAFkTCPLeX8u0V"

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

```

→ go to container → IAM

→ not above

Home > olidatastoragedinedesh | Containers > olidata

olidata | Access Control (IAM)

Container

Search Overview Download role assignments Edit columns Refresh Delete Feedback

Check access Role assignments Roles Deny assignments Classic administrators

My access View my level of access to this resource.

Check access Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

Check access

Grant access to this resource Add role assignment

View access to this resource View

View deny assignments View

New! Permissions Management Discover, monitor and remediate unused

→ click Add → Add assign role

Home > olidatastoragedinedesh | Containers > olidata | Access Control (IAM)

Add role assignment

Role Members Conditions Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Job function roles Privileged administrator roles

Grant access to Azure resources based on job function, such as the ability to create virtual machines.

Name	Description	Type	Category	Details
Defender CSPM Storage Data Scanner	Grants access to read blobs and files. This role is used by the data scanner of Defender CSPM.	BuiltinRole	None	View
Defender for Storage Data Scanner	Grants access to read blobs and update index tags. This role is used by the data scanner of Defender for Storage.	BuiltinRole	None	View
Storage Blob Data Contributor	Allows for read, write and delete access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Data Owner	Allows for full access to Azure Storage blob containers and data, including assigning POSIX access control.	BuiltinRole	Storage	View
Storage Blob Data Reader	Allows for read access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Delegator	Allows for generation of a user delegation key which can be used to sign SAS tokens	BuiltinRole	Storage	View

Showing 1 - 6 of 6 results.

Review + assign Previous Next Feedback

→ click user group or principle → identity and select members

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

Review + assign Previous Next Feedback

Home > App registrations

+ New registration Endpoints Troubleshoot Refresh Download Preview features Got feedback?

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

All applications Owned applications Deleted applications Applications from personal account

Start typing a display name or application (client) ID to filter these results Add filters

Display name	Application (client) ID	Created on	Certificates & secrets
olist-app-registration-db-adfs	260e6732-04d0-4d7c-be5d-fba04b30aa93	3/19/2025	Current

→ same name in select members

Microsoft Azure

Home > olistdatastoragedinedesh | 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

Review + assign Previous Next Select Close

Select members

olist-app-registration-db-adfs

olist-app-registration-db-adfs Application

Selected members:

No members selected. Search for and add one or more members you want to assign to the role for this resource.

Learn more about RBAC

Microsoft Azure

Home > olistdatastoragedinedesh | 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

Review + assign Previous Next Select Close

Select members

olist-app-registration-db-adfs

olist-app-registration-db-adfs Application

Selected members:

olist-app-registration-db-adfs Application

Select Close

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

dineshkumar7283@g...
DEFAULT DIRECTORY

Home > olistdatastorage | Containers > olistdata

olistdata | Access Control (IAM)

Container

Search

Add Download role assignments Edit columns Refresh Delete Feedback

Overview Diagnose and solve problems

Access Control (IAM)

Settings

Check access Role assignments Roles Deny assignments Classic administrators

My access

View my level of access to this resource.

[View my access](#)

Check access

Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

[Check access](#)

Grant access to this resource

Grant access to resources by assigning a role. [Learn more](#)

[Add role assignment](#)

View access to this resource

View the role assignments that grant access to this and other resources. [Learn more](#)

[View](#)

View deny assignments

View the role assignments that have been denied access to specific actions at this scope. [Learn more](#)

[View](#)

New! Permissions Management

Discover, monitor and remediate unused permissions in your Azure environment with Microsoft Estate Permissions Management

Added Role assignment
olist-app-registration-db-adfs was added as Storage Blob Data Contributor for olistdata.

→ verify again

Microsoft Azure Search resources, services, and docs (G+) Copilot

Home > Resource groups > Ecomm-live > olistdatastorage | Access Control (IAM) > Add role assignment

Add role assignment

Role Members Conditions Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Job function roles Privileged administrator roles

Grant access to Azure resources based on job function, such as the ability to create virtual machines.

Name ↑↓	Description ↑↓	Type ↑↓	Category ↑↓	Details
Defender CSPM Storage Data Scanner	Grants access to read blobs and files. This role is used by the data scanner of Defender CSPM.	BuiltinRole	None	View
Defender for Storage Data Scanner	Grants access to read blobs and update index tags. This role is used by the data scanner of Defender for Storage.	BuiltinRole	None	View
Storage Blob Data Contributor	Allows for read, write and delete access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Data Owner	Allows for full access to Azure Storage blob containers and data, including assigning POSIX access control.	BuiltinRole	Storage	View
Storage Blob Data Reader	Allows for read access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Delegator	Allows for generation of a user delegation key which can be used to sign SAS tokens	BuiltinRole	Storage	View

Showing 1 - 6 of 6 results.

[Review + assign](#) [Previous](#) [Next](#)

Microsoft Azure

Home > Resource groups > Ecomm-live > olistdatastoragedinesh | 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

Review + assign Previous Next Select Close

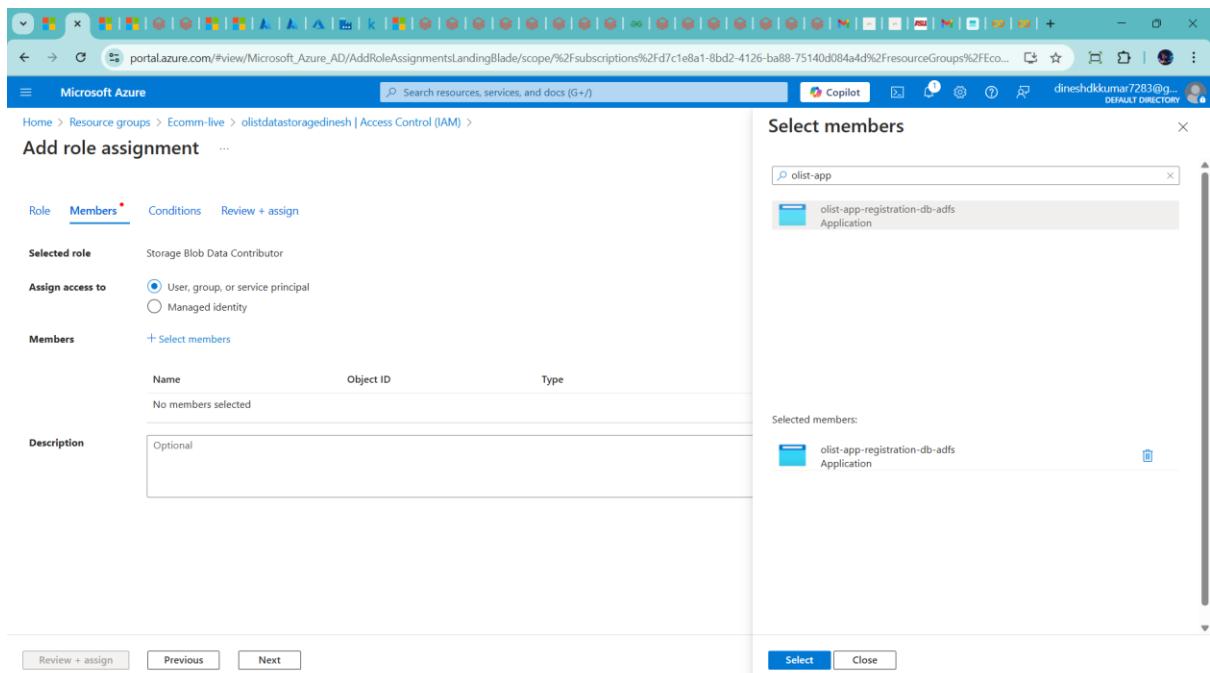
Select members

olist-app

olist-app-registration-db-adfs Application

Selected members:

olist-app-registration-db-adfs Application



Microsoft Azure databricks

Untitled Notebook 2025-03-19 11:03:54 Python

New Workspace Recents Catalog Workflows Compute Marketplace

SQL SQL Editor Queries Dashboards Genie Alerts Query History SQL Warehouses

Data Engineering Job Runs Data Ingestion Pipelines Machine Learning Playground Experiments

File Edit View Run Help Last edit was now

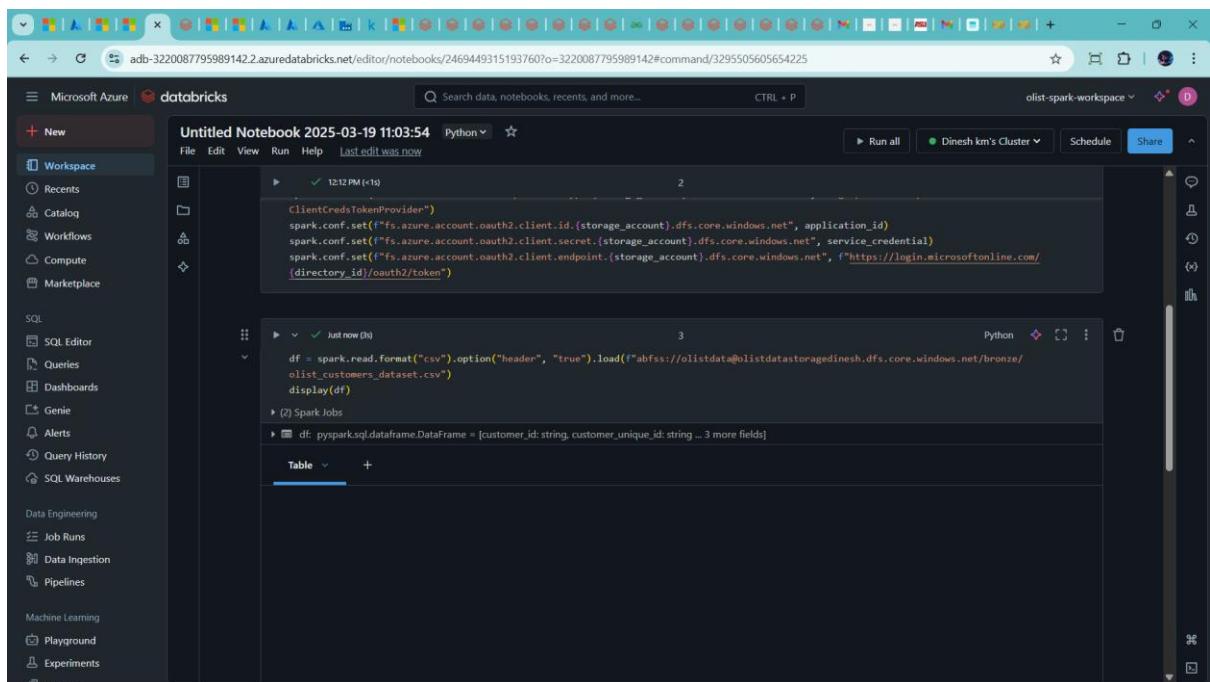
Run all Dinesh km's Cluster Schedule Share

```
ClientCredsTokenProvider")
spark.conf.set("fs.azure.account.oauth2.client.id.(storage_account).dfs.core.windows.net", application_id)
spark.conf.set("fs.azure.account.oauth2.client.secret.(storage_account).dfs.core.windows.net", service_credential)
spark.conf.set("fs.azure.account.oauth2.client.endpoint.(storage_account).dfs.core.windows.net", "https://login.microsoftonline.com/
(directory_id)/oauth2/token")
```

Just now (3) df = spark.read.format("csv").option("header", "true").load("abfss://olistdata@olistdatastoragedinesh.dfs.core.windows.net/bronze/
olist_customers_dataset.csv")
display(df)

(2) Spark Jobs df: pyspark.sql.dataframe.DataFrame = [customer_id:string, customer_unique_id:string ... 3 more fields]

Table +



Part--2

The screenshot shows a Databricks notebook interface. The sidebar on the left contains various navigation links such as Workspace, Recents, Catalog, Workflows, Compute, Marketplace, SQL, and Data Engineering. The main area displays a notebook titled "Untitled Notebook 2025-03-19 11:03:54" in Python. The code cell contains the following Python code:

```
> customers_df = spark.read\...
> geolocations_df = spark.read\...
> items_df = spark.read\...
> payments_df = spark.read\...
> reviews_df = spark.read\...
> orders_df = spark.read\...
> products_df = spark.read\...
sellers_df = spark.read\
    .format("csv")\
    .option("header", "true")\
    .option("inferSchema", "true")\
    .load("abfss://olistdatastoragedinedh.dfs.core.windows.net/bronze/olist_sellers_dataset.csv")
```

→they show error

The screenshot shows a Databricks notebook interface. The sidebar on the left contains various navigation links such as Workspace, Recents, Catalog, Workflows, Compute, Marketplace, SQL, and Data Engineering. The main area displays a notebook titled "Untitled Notebook 2025-03-19 11:03:54" in Python. The code cell contains the following Python code:

```
> customers_df: pyspark.sql.DataFrame = [customer_id: string, customer_unique_id: string ... 3 more fields]
> geolocations_df: pyspark.sql.DataFrame = [geolocation_zip_code_prefix: integer, geolocation_lat: double ... 5 more fields]
> items_df: pyspark.sql.DataFrame = [order_id: string, order_item_id: integer ... 5 more fields]
> orders_df: pyspark.sql.DataFrame = [order_id: string, customer_id: string ... 6 more fields]
> payments_df: pyspark.sql.DataFrame = [order_id: string, payment_sequential: integer ... 3 more fields]
> products_df: pyspark.sql.DataFrame = [product_id: string, product_category_name: string ... 7 more fields]
> reviews_df: pyspark.sql.DataFrame = [review_id: string, order_id: string ... 5 more fields]
> sellers_df: pyspark.sql.DataFrame = [seller_id: string, seller_zip_code_prefix: integer ... 2 more fields]
```

Below the code cell, an error message is displayed:

Last execution failed
1 import pymongo
2 ModuleNotFoundError: No module named 'pymongo'

→go to compute→click libraries

The screenshot shows the Databricks Compute Libraries page for a cluster named "Dinesh km's Cluster". The left sidebar is visible with various navigation options like Workspace, Catalog, Workflows, Compute, and SQL. The main area has tabs for Configuration, Notebooks (1), Libraries, Event log, Spark UI, Driver logs, Metrics, Apps, and Spark compute UI - Master. The Libraries tab is selected. A search bar at the top says "Search data, notebooks, recents, and more...". Below it, there's a "Filter libraries" input field. To the right are "Terminate" and "Edit" buttons. A table header includes columns for Status, Name, Type, and Source. A message in the center says "No libraries" and "Please install new libraries with Install New".

→click install new →click pypi→type pymongo →click install

The screenshot shows the "Install library" dialog box overlaid on the Compute Libraries page. The dialog has a title "Install library" with a "Send feedback" link. It includes a "Library Source" section with radio buttons for Workspace, Volumes, File Path/ADLS, PyPi (which is selected), Maven, and CRAN. A warning message says "We recommend specifying an exact version of the library with == to prevent regressions. Learn more". Below that is a "Package" input field containing "Pymongo", with a note "Warning: package is not pinned". There's also an "Index URL" input field with "Optional" selected. At the bottom are "Cancel" and "Install" buttons.

The screenshot shows the Databricks interface. On the left, there's a sidebar with various navigation options like Workspace, Catalog, Workflows, Compute, SQL, Data Engineering, etc. The main area is titled 'Dinesh km's Cluster' and shows the 'Libraries' tab selected. A table lists one library: 'pymongo' (Type: PyPI). There are buttons for 'Uninstall' and 'Install new'. At the top right, there are 'Terminate' and 'Edit' buttons.

→ go workspace in notebook run again

The screenshot shows a Databricks notebook titled 'Untitled Notebook 2025-03-19 11:03:54'. The sidebar on the left is identical to the previous screenshot. The notebook content includes a code cell with the following Python code:`from pymongo import MongoClient`

Below the code cell, there's a message: 'Start typing on generate with AI (Ctrl + I)...'. The notebook interface includes standard Databricks controls like 'Run all', 'Schedule', and 'Share'.

→ go to files.io in mongodb

The screenshot shows the filess.io dashboard interface. On the left, there's a sidebar with user information (dinesh km, dineshkumar63519, 500@gmail.com, Free tier), navigation links (Databases, Standard Tier, Billing, API, Status), and a 'Get more?' section with an 'Upgrade Now' button. The main area displays a database named 'olistDataNoSQL'. It features a large green shield icon. Below the icon, it says 'Mongo v7.0.2 • Status Up'. There are two buttons: 'Web Client' and 'Code'. The 'Code' button is highlighted. To its right is a 'Connection formats' section with a MongoDB URI: 'mongodb://olistDataNoSQL_poolcanpie:...@9o6fs.h.filess.io:27018/olistDataNoSQL_poolca'. At the top right of the main area are icons for Smart Query, a bell (with 3 notifications), and a user profile.

→click code→seach python and copy

This screenshot shows the 'Generate code' page for the 'olistDataNoSQL' database. The left sidebar is identical to the previous dashboard. The main area has a heading 'Generate code' with a note: 'Generate code to connect to your database. You can use this code in your application.'. A dropdown menu 'Language' is set to 'Python'. Below it is a code editor containing Python code to connect to the database:

```
1 # importing module
2 from pymongo import MongoClient
3
4 hostname = "9o6fs.h.filess.io"
5 database = "olistDataNoSQL_poolcanpie"
6 port = "27018"
7 username = "olistDataNoSQL_poolcanpie"
8 password = "fa9f830f3ca3365abde1af4a10a4041c522fc3"
9
10 uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database
11
12 # Connect with the portnumber and host
13 client = MongoClient(uri)
14
15 # Access database
16 mydatabase = client[database]
```

A yellow bar at the bottom of the code editor contains a 'Copy Code' button. A green checkmark icon and the text 'Code copied to clipboard' are displayed above the code editor. At the bottom of the page, there's a note: 'If you want to contribute to this project and add support for your language, you can create a pull request in our [GitHub repository](#)'.

→add in notebook

Untitled Notebook 2025-03-19 11:03:54 Python

```
from pymongo import MongoClient

hostname = "9o6fs.h.fileless.io"
database = "olistDataNoSQL_poolcanpie"
port = "27018"
username = "olistDataNoSQL_poolcanpie"
password = "fa9f830f3ca3365abdeafdfdd10a4041c522fc3"

uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database

# Connect with the portnumber and host
client = MongoClient(uri)

# Access database
mydatabase = client[database]
```

Untitled Notebook 2025-03-19 11:03:54 Python

```
# Importing module
from pymongo import MongoClient

hostname = "9o6fs.h.fileless.io"
database = "olistDataNoSQL_poolcanpie"
port = "27018"
username = "olistDataNoSQL_poolcanpie"
password = "fa9f830f3ca3365abdeafdfdd10a4041c522fc3"

uri = "mongodb://" + username + ":" + password + "@" + hostname + ":" + port + "/" + database

# Connect with the portnumber and host
client = MongoClient(uri)

# Access database
mydatabase = client[database]
```

Type: Database
String form: pymongo.synchronous.database.Database instance
Docstring: <no docstring>

```
mydatabase
Database(MongoClient(host=['9o6fs.h.fileless.io:27018'], document_class=dict, tz_aware=False, connect=True), 'olistDataNoSQL_poolcanpie')
```

Untitled Notebook 2025-03-19 11:03:54 Python

```
Database(MongoClient(host=['9o6fs.h.filess.io:27018']), document_class=dict, tz_aware=False, connect=True, 'olistDataNoSQL_poolcanpie')

collection = database['product_category_name_translation']
mongo_df = pd.DataFrame(list(collection.find()))
mongo_df
```

_id	product_category_name	product_category_name_english
0	beleza_saude	health_beauty
1	informatica_acessorios	computers_accessories
2	automotvo	auto
3	cama_mesa_banho	bed_bath_table
4	moveis_decoracao	furniture_decor
...
66	flores	flowers
67	artes_e_artesanato	arts_and_craftsmanship
68	fraldas_higene	diapers_and_hygiene
69	fashion_roupa_infantil_juvenil	fashion_childrens_clothes
70	seguros_e_servicos	security_and_services

71 rows × 3 columns

Cleaning the data

```
from pyspark.sql.functions import col,to_date,datediff,current_date

def clean_dataframe(df,name):
    print("Cleaning "+name)
    return df.dropDuplicates().na.drop('all')

orders_df=clean_dataframe(orders_df,"Orders")
display(orders_df)
```

Cleaning Orders

order_id	customer_id	order_status	order_purchase_timestamp	order_approved_at
4	a9:9532060c9d245106526c633d2dfba	delivered	2018-01-02T19:20:35.000+00:00	2018-01-02T19:32:22.000+00:00
5	ca90a0ee09415b5f5679c93b5191633	acd575d7382968889f1a5a3f57510dd	delivered	2018-04-09T22:02:30.000+00:00
6	37357208c1cd4212fa0866a8e58a049	94af59d9cac1ae1976312584f628ef6f	delivered	2018-05-16T23:08:51.000+00:00
7	8b346abc3486e64bc67fc3b0ec0cd9f	dd1506d329d9b0135855082aae3c79ac	delivered	2018-07-11T20:15:04.000+00:00
8	a4d7c88ca45b56444e3c59ddcca7d7c9	fdcb673d0f8d2b84d3862193f17a089	delivered	2017-08-30T02:05:44.000+00:00

Untitled Notebook 2025-03-19 11:03:54 Python

```
File Edit View Run Help Last edit was now
from pyspark.sql.functions import col,to_date,datediff,current_date,when
def clean_dataframe(df,name):
    print("Cleaning "+name)
    return df.dropDuplicates().na.drop('all')

orders_df=clean_dataframe(orders_df,"Orders")
display(orders_df)
```

Cleaning the data

	order_id	customer_id	order_purchase_timestamp	order_delivered_customer_date	order_estimated_delivery_date	actual_delivery_time	estimated_delivery_time	delay
1	67da5524c9281859c7897fb	traias_nigiene	2018-06-16T15:20:55.000+0000	2018-07-18T00:00:00.000+0000		12	44	0
2	67da5524c9281859c7897fb	fashion_rroupa_infant_juvenil	2018-03-09T21:52:36.000+0000	2018-03-09T00:00:00.000+0000		23	23	0
3	67da5524c9281859c7897fb	seguros_e_servicos	2018-02-07T00:00:00.000+0000	2018-02-07T00:00:00.000+0000		null	28	0
4	67da5524c9281859c7897fb		2018-01-27T14:27:59.000+0000	2018-02-05T00:00:00.000+0000		25	34	0
5	67da5524c9281859c7897fb		2018-04-25T13:33:24.000+0000	2018-05-10T00:00:00.000+0000		16	31	0
6	67da5524c9281859c7897fb		2018-05-21T20:50:54.000+0000	2018-06-20T00:00:00.000+0000		5	35	0
7	67da5524c9281859c7897fb		2018-07-26T23:32:50.000+0000	2018-08-09T00:00:00.000+0000		15	29	0
8	67da5524c9281859c7897fb		2017-09-08T21:36:28.000+0000	2017-09-22T00:00:00.000+0000		10	24	0
9	67da5524c9281859c7897fb		2017-05-24T15:06:39.000+0000	2017-06-21T00:00:00.000+0000		18	46	0
10	67da5524c9281859c7897fb		2018-05-28T16:46:42.000+0000	2018-06-13T00:00:00.000+0000		4	20	0
11	67da5524c9281859c7897fb		2017-04-04T15:49:42.000+0000	2017-04-18T00:00:00.000+0000		15	29	0

Untitled Notebook 2025-03-19 11:03:54 Python

```
File Edit View Run Help Last edit was now
#Calculate delivery' and time delays
orders_df = orders_df.withColumn("actual_delivery_time",datediff("order_delivered_customer_date", "order_purchase_timestamp"))
orders_df=orders_df.withColumn("estimated_delivery_time",datediff("order_estimated_delivery_date",'order_purchase_timestamp'))
orders_df=orders_df.withColumn("delay",when(col("actual_delivery_time")>col("estimated_delivery_time"),1).otherwise(0))

display(orders_df)
```

	order_id	customer_id	order_purchase_timestamp	order_delivered_customer_date	order_estimated_delivery_date	actual_delivery_time	estimated_delivery_time	delay
1	67da5524c9281859c7897fb	traias_nigiene	2018-06-16T15:20:55.000+0000	2018-07-18T00:00:00.000+0000		12	44	0
2	67da5524c9281859c7897fb	fashion_rroupa_infant_juvenil	2018-03-09T21:52:36.000+0000	2018-03-09T00:00:00.000+0000		23	23	0
3	67da5524c9281859c7897fb	seguros_e_servicos	2018-02-07T00:00:00.000+0000	2018-02-07T00:00:00.000+0000		null	28	0
4	67da5524c9281859c7897fb		2018-01-27T14:27:59.000+0000	2018-02-05T00:00:00.000+0000		25	34	0
5	67da5524c9281859c7897fb		2018-04-25T13:33:24.000+0000	2018-05-10T00:00:00.000+0000		16	31	0
6	67da5524c9281859c7897fb		2018-05-21T20:50:54.000+0000	2018-06-20T00:00:00.000+0000		5	35	0
7	67da5524c9281859c7897fb		2018-07-26T23:32:50.000+0000	2018-08-09T00:00:00.000+0000		15	29	0
8	67da5524c9281859c7897fb		2017-09-08T21:36:28.000+0000	2017-09-22T00:00:00.000+0000		10	24	0
9	67da5524c9281859c7897fb		2017-05-24T15:06:39.000+0000	2017-06-21T00:00:00.000+0000		18	46	0
10	67da5524c9281859c7897fb		2018-05-28T16:46:42.000+0000	2018-06-13T00:00:00.000+0000		4	20	0
11	67da5524c9281859c7897fb		2017-04-04T15:49:42.000+0000	2017-04-18T00:00:00.000+0000		15	29	0

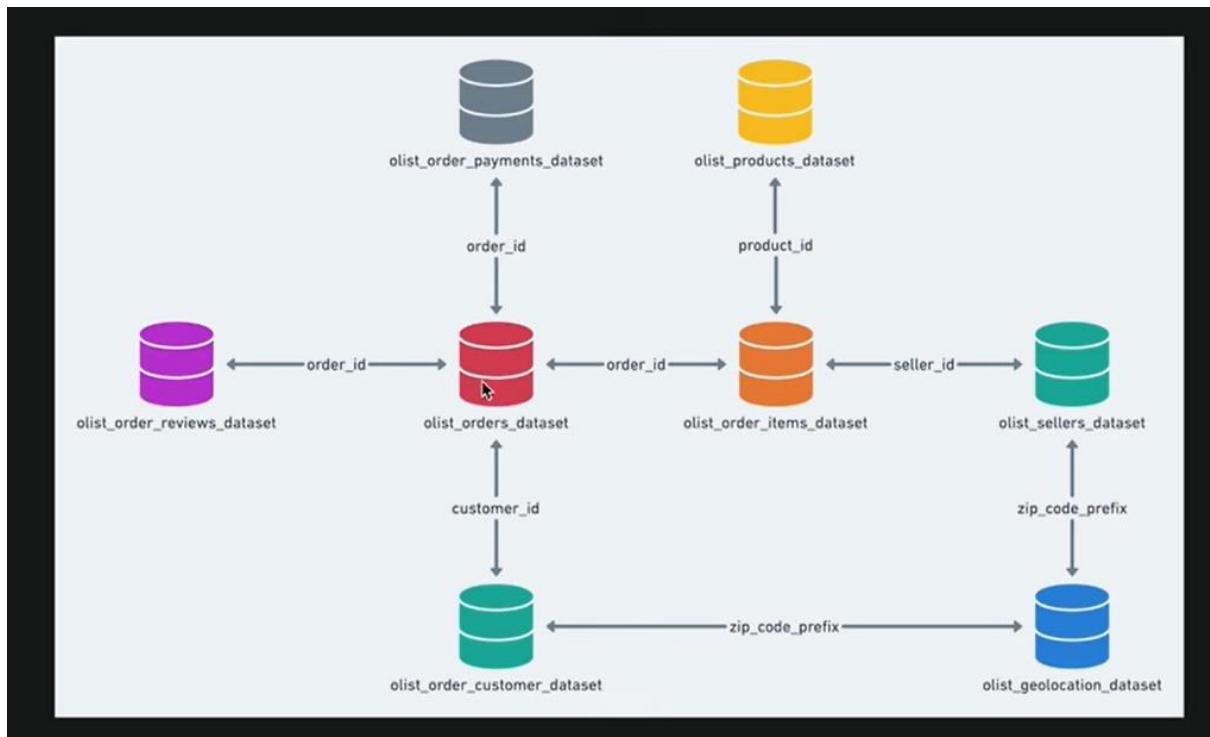
Untitled Notebook 2025-03-19 11:03:54 Python

```
# Calculate delivery and time delays
orders_df = orders_df.withColumn("actual_delivery_time", datediff("order_delivered_customer_date", "order_purchase_timestamp"))
orders_df = orders_df.withColumn("estimated_delivery_time", datediff("order_estimated_delivery_date", "order_purchase_timestamp"))
orders_df = orders_df.withColumn("Delay Time", col("actual_delivery_time") - col("estimated_delivery_time"))

display(orders_df)
```

	red_customer_date	order_estimated_delivery_date	actual_delivery_time	estimated_delivery_time	delay	Delay Time
1	1055.000+0000	2018-07-18T00:00:00.000+0000	12	44	false	-32
2	1236.000+0000	2018-03-09T00:00:00.000+0000	23	23	false	0
3		2018-02-07T00:00:00.000+0000	[null]	28	[null]	[null]
4	1759.000+0000	2018-02-05T00:00:00.000+0000	25	34	false	-9
5	1324.000+0000	2018-05-10T00:00:00.000+0000	16	31	false	-15
6	1054.000+0000	2018-06-20T00:00:00.000+0000	5	35	false	-30
7	1250.000+0000	2018-08-09T00:00:00.000+0000	15	29	false	-14
8	1628.000+0000	2017-09-22T00:00:00.000+0000	10	24	false	-14
9	1639.000+0000	2017-06-21T00:00:00.000+0000	18	46	false	-28
10	1642.000+0000	2018-06-13T00:00:00.000+0000	4	20	false	-16
11	1942.000+0000	2017-04-18T00:00:00.000+0000	15	29	false	-14
12	1224.000+0000	2018-08-17T00:00:00.000+0000	6	10	false	-4
13	1038.000+0000	2018-02-15T00:00:00.000+0000	2	16	false	-14
14	1833.000+0000	2018-09-07T00:00:00.000+0000	8	36	false	-28

→Joining



adb-3220087795989142.azuredatabricks.net/editor/notebooks/2469449315193760?o=3220087795989142

Untitled Notebook 2025-03-19 11:03:54 Python

Last edit was now

File Edit View Run Help Refreshed 5 minutes ago

Run all Dinesh km's Cluster Schedule Share

Joining

```
Just now (<1s)
orders_customer_df=orders_df.join(customers_df,orders_df.customer_id==customers_df.customer_id,how='left')
orders_payment_df=orders_customer_df.join(payments_df,orders_customer_df.order_id==payments_df.order_id,how='left')
orders_items_df=orders_payment_df.join(items_df,"order_id",how='left')
orders_item_product_df=orders_items_df.join(products_df,orders_items_df.product_id==products_df.product_id,how='left')
final_df=orders_item_product_df.join(sellers_df,orders_item_product_df.seller_id==sellers_df.seller_id,how='left')

final_df=pyspark.sql.DataFrame = [order_id:string, customer_id:string ... 38 more fields]
orders_customer_df: pyspark.sql.DataFrame = [order_id:string, customer_id:string ... 14 more fields]
orders_items_df: pyspark.sql.DataFrame = [order_id:string, customer_id:string ... 34 more fields]
orders_items_df: pyspark.sql.DataFrame = [order_id:string, customer_id:string ... 25 more fields]
orders_payment_df: pyspark.sql.DataFrame = [order_id:string, customer_id:string ... 19 more fields]
```

```
Just now (5s)
display(final_df)
(7) Spark Jobs
```

```
Start typing or generate with AI (Ctrl + I)...
```

adb-3220087795989142.azuredatabricks.net/editor/notebooks/2469449315193760?o=3220087795989142#command/4462995065565000

Untitled Notebook 2025-03-19 11:03:54 Python

Last edit was now

File Edit View Run Help Refreshed 5 minutes ago

Run all Dinesh km's Cluster Schedule Share

	19	88bf34949a5bb708a123dd12e80ed24d	7dd85f58cadcc624d2f05f6a2049dc04	delivered	2018-07-03T23:02:42.000+00:00	2018-07-05T16:26:57.000+00:00
	20	98826cd8a97e41725b525e492a8d7571	4415b9cc1e718722dd4b4a79ccb08e0	delivered	2017-05-20T15:38:28.000+00:00	2017-05-20T15:50:08.000+00:00
	21	d43e09377465f739e75e7e7a51645a	2d1cc6fad845ab8d3389a26d9c63b29	delivered	2017-05-25T16:20:36.000+00:00	2017-05-25T16:30:15.000+00:00
	22	17df2cc606801c1712991499b645cd2	41fd87556cc2ddee491b0cb9e05f155	delivered	2017-05-01T18:58:53.000+00:00	2017-05-01T18:58:53.000+00:00
	23	5b7f6445ca01d94adfe256eaa9a37ada	202a41667703ebcde9400559685645	delivered	2017-09-19T20:18:47.000+00:00	2017-09-19T20:30:12.000+00:00
	24	5b0ee19a3125fc0833abaa7d816ed0	97e5ea711c00ba5439zbfb66a7d7f1e0	canceled	2018-03-05T13:56:34.000+00:00	2018-03-05T14:40:27.000+00:00

3,394+ rows | Truncated data | 4:73s runtime

```
Just now (1s)
mongo_df.drop('_id',axis=1,inplace=True)
mongo_spark_df=pyspark.createDataFrame(mongo_df)

mongo_spark_df: pyspark.sql.DataFrame = [product_category_name:string, product_category_name_english:string]
```

```
Just now (<1s)
final_df=final_df.join(mongo_spark_df, on='product_category_name', how='left')

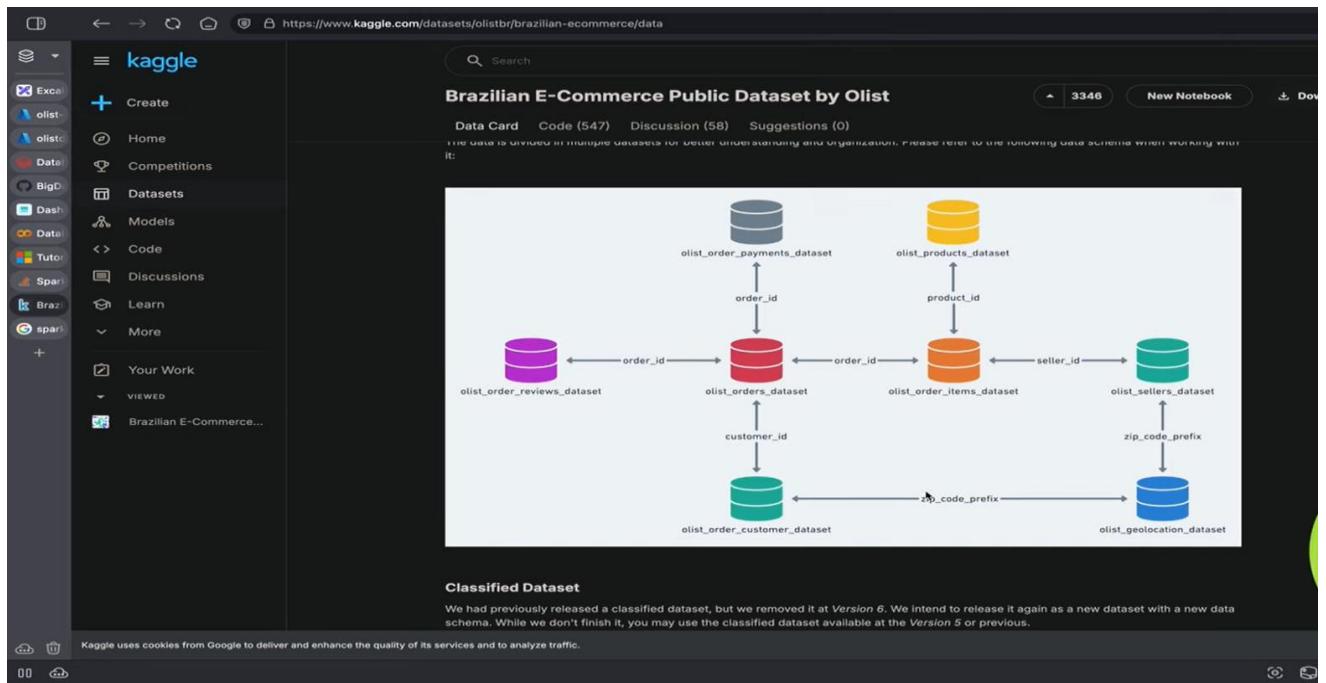
final_df=pyspark.sql.DataFrame = [product_category_name:string, order_id:string ... 39 more fields]
```

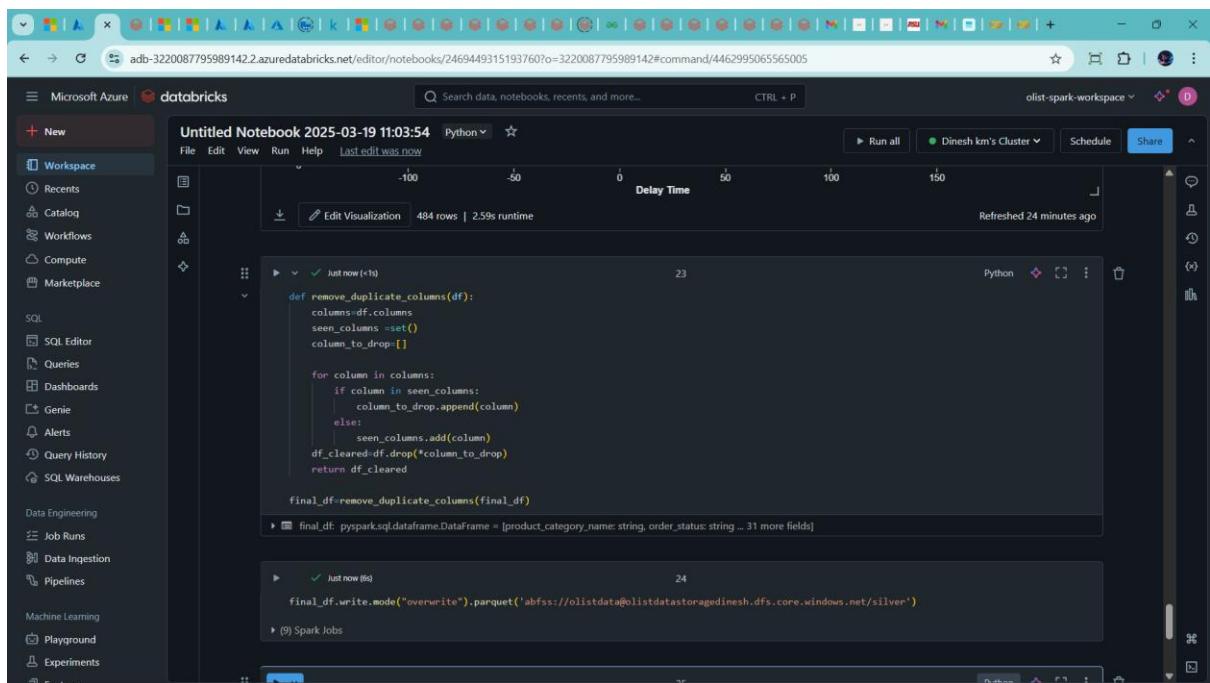
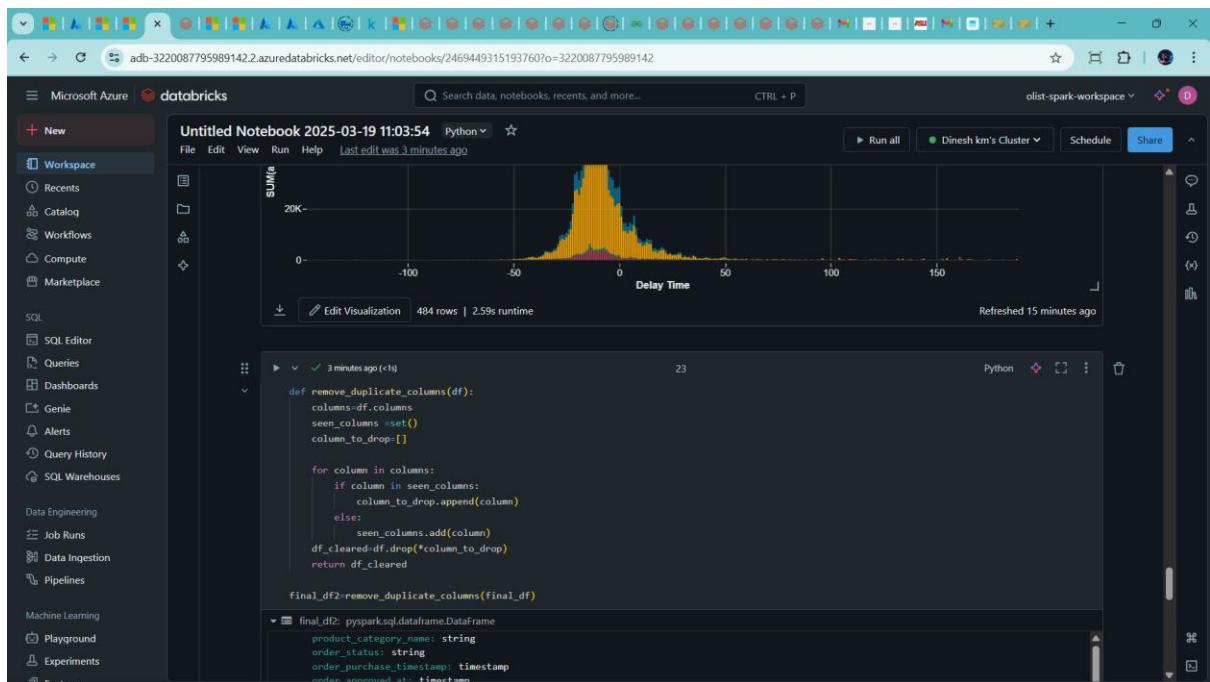
```
Start typing or generate with AI (Ctrl + I)...
```

[Shift+Enter] to run and move to next cell
[Ctrl+Shift+F9] to open the command palette

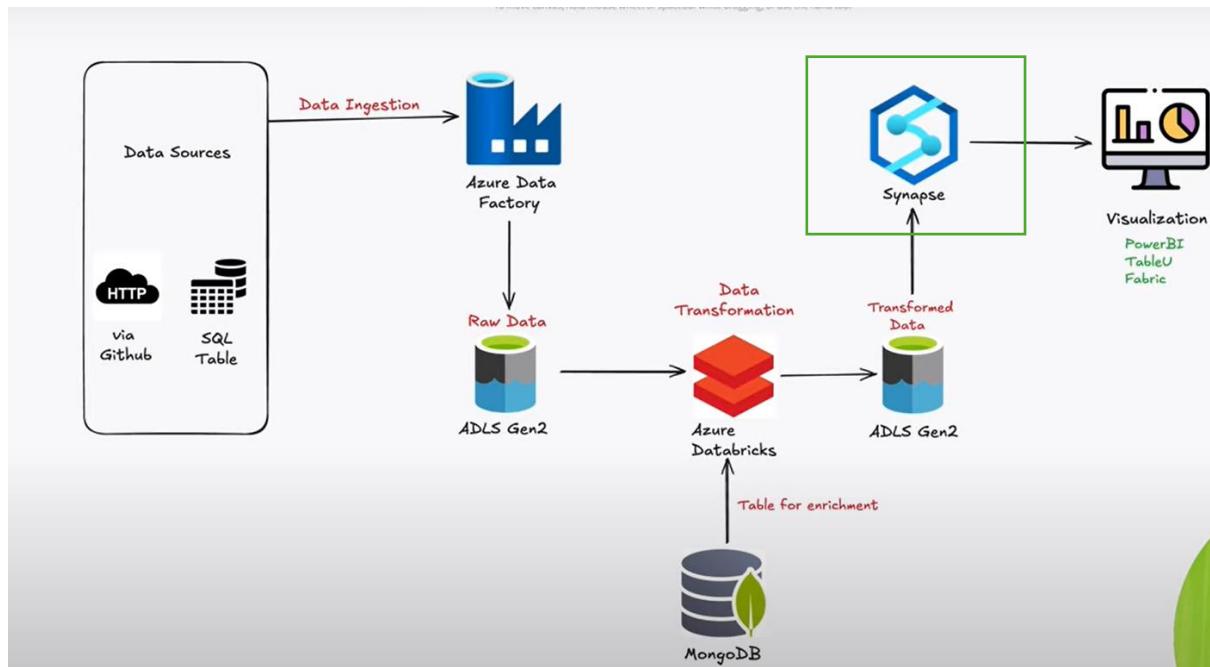


This is over dataset





Synapse



→create workspace synapse in azure

The screenshot shows the Microsoft Azure portal interface. The URL is <https://portal.azure.com/#@dineshdkumar7283@gmail.onmicrosoft.com/resource/subscriptions/d7c1e8a1-8bd2-4126-ba88-75140d084a4d/resourceGroups/Ecomm-live/overview>. The page displays the 'Ecomm-live' resource group details, including its subscription information (Subscription (move) : Azure for Students, Subscription ID : d7c1e8a1-8bd2-4126-ba88-75140d084a4d, Tags (edit) : Add.tags), deployment status (Deployments : 3 Succeeded), and location (Location : Central India). The 'Resources' section lists three resources: 'olist-ecomm-data-factory-dinesh' (Data factory (V2)), 'olist-spark-workspace' (Azure Databricks Service), and 'olistdatastoragedinesh' (Storage account). The page also includes navigation links like 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Resource visualizer', 'Events', 'Settings', 'Cost Management', 'Monitoring', 'Automation', and 'Help'. At the bottom, there are pagination controls ('Page 1 of 1') and a 'Give feedback' link.

The screenshot shows the Microsoft Azure Marketplace page for Azure Synapse Analytics. At the top, there's a search bar and a Copilot button. The main heading is "Azure Synapse Analytics". Below it, there's a brief description: "Azure Synapse is a limitless analytics service that brings together data integration, enterprise data warehousing, and Big Data analytics." It highlights the freedom to query data on terms, using either serverless or dedicated resources at scale. Key service capabilities include a unified analytics platform, serverless and dedicated options, enterprise data warehouse, data lake exploration, code-free hybrid data integration, deeply integrated Apache Spark and SQL engines, cloud-native HTAP, choice of language (T-SQL, Python, Scala, SparkSQL, & .NET), and integrated AI and BI.

→ synapse manage different storage taken

The screenshot shows the "Create Synapse workspace" wizard. It starts with "Subscription" set to "Azure for Students", "Resource group" set to "Ecomm-live", and "Managed resource group" set to "olidata-synapse-rg". The "Workspace details" section asks for a name ("olist-synapse-workspace-dinesh"), region ("Central India"), and a Data Lake Storage Gen2 account ("(New) synapselfilesys"). A note says "Assign myself the Storage Blob Data Contributor role on the Data Lake Storage Gen2 account to interactively query it in the workspace." Navigation buttons at the bottom are "Review + create", "< Previous", and "Next: Security >".

→click new

Configure security options for your workspace.

Authentication

Choose the authentication method for access to workspace resources such as SQL pools. The authentication method can be changed later on. [Learn more](#)

Authentication method Use both local and Microsoft Entra ID authentication Use only Microsoft Entra ID authentication

SQL Server admin login *

SQL Password

Confirm password

Your login name must not contain a SQL Identifier or a typical system name (like admin, administrator, sa, root, dbmanager, loginmanager, etc) or a built-in database user or role (like dbo, guest, public, etc)

Your login name must not include non-alphanumeric characters.

Your login name must be between 1 and 128 characters long.

Your login name must not start with numbers or symbols

System assigned managed identity permission

Select to grant the workspace network access to the Data Lake Storage Gen2 account using the workspace system identity. [Learn more](#)

Allow network access to Data Lake Storage Gen2 account. network access using any network access rules, or you selected a storage account manually via URL under Basics tab. [Learn more](#)

Workspace encryption

⚠ Double encryption configuration cannot be changed after opting into using a customer-managed key at the time of workspace creation.

Review + create < Previous Next: Networking >

→click new

Configure networking options for your workspace.

Managed virtual network

Choose whether to set up a dedicated Azure Synapse-managed virtual network for your workspace. [Learn more](#)

Managed virtual network Enable Disable

To control public network access to your Synapse workspace, you must enable managed virtual network.

Firewall rules

Azure Synapse Studio and other client tools will only connect to the workspace endpoints if this setting is selected. Connections from specific IP addresses or all Azure services can be allowed or disallowed after the workspace is provisioned.

Allow connections from all IP addresses

Encrypted connections

Azure Synapse Analytics applies TLS1.2 encryption protocol for all network connections. [Learn more](#)

Minimum TLS version

Review + create < Previous Next: Tags >

→if enable you have sensitive data

So by default →click new and review + create

Validation succeeded

Product Details

Azure Synapse Analytics workspace
by Microsoft

Serverless SQL est. cost/TB 415.97 INR

Terms

By clicking Create, I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see [Azure Marketplace Terms](#).

Basics

Subscription	Azure for Students
Resource group	Ecomm-live
Region	Central India
Workspace name	(new) olist-synapse-workspace-dinesh
Data Lake Storage Gen2 account	(new) https://synaptestoragedinesh.dfs.core.windows.net

Create < Previous Next > Download a template for automation

→complete

olist-synapse-workspace-dinesh

Synapse workspace

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource visualizer

Settings

Analytics pools

Security

Monitoring

Automation

Help

Getting started

Open Synapse Studio

Start building your fully-integrated analytics solution and unlock new insights.

Read documentation

Learn how to be productive quickly. Explore concepts, tutorials, and samples.

Notifications

More events in the activity log → Dismiss all

Deployment succeeded Deployment 'Microsoft.Azure.SynapseAnalytics-20250319151545' to resource group 'Ecomm-live' was successful.

Pin to dashboard Go to resource group a few seconds ago

Added Role assignment olist-app-registration-db-adfs was added as Storage Blob Data Contributor for olistdatastoragedinesh.

Deployment succeeded Deployment 'Ecomm-live_olist-spark-workspace' to resource group 'Ecomm-live' was successful.

Go to resource Pin to dashboard 5 hours ago

→click open

olist-synapse-workspace-dinesh

Overview

Location: Central India
Subscription: Azure for Students
Subscription ID: d7c1e8a1-8bd2-4126-ba88-75140d084a4d
Tags: olist-synapse-workspace-dinesh
Resource visualizer: olist-synapse-workspace-dinesh
Getting started: Open Synapse Studio, Read documentation

Analytics pools

Notifications

More events in the activity log → Dismiss all

Deployment succeeded Deployment 'Microsoft.Azure.SynapseAnalytics-20250319151545' to resource group 'Ecomm-live' was successful.

Added Role assignment olist-app-registration-db-adfs was added as Storage Blob Data Contributor for olistdatastorageadfdinesh.

Deployment succeeded Deployment 'Ecomm-live_olist-spark-workspace' to resource group 'Ecomm-live' was successful.

Azure Synapse Analytics

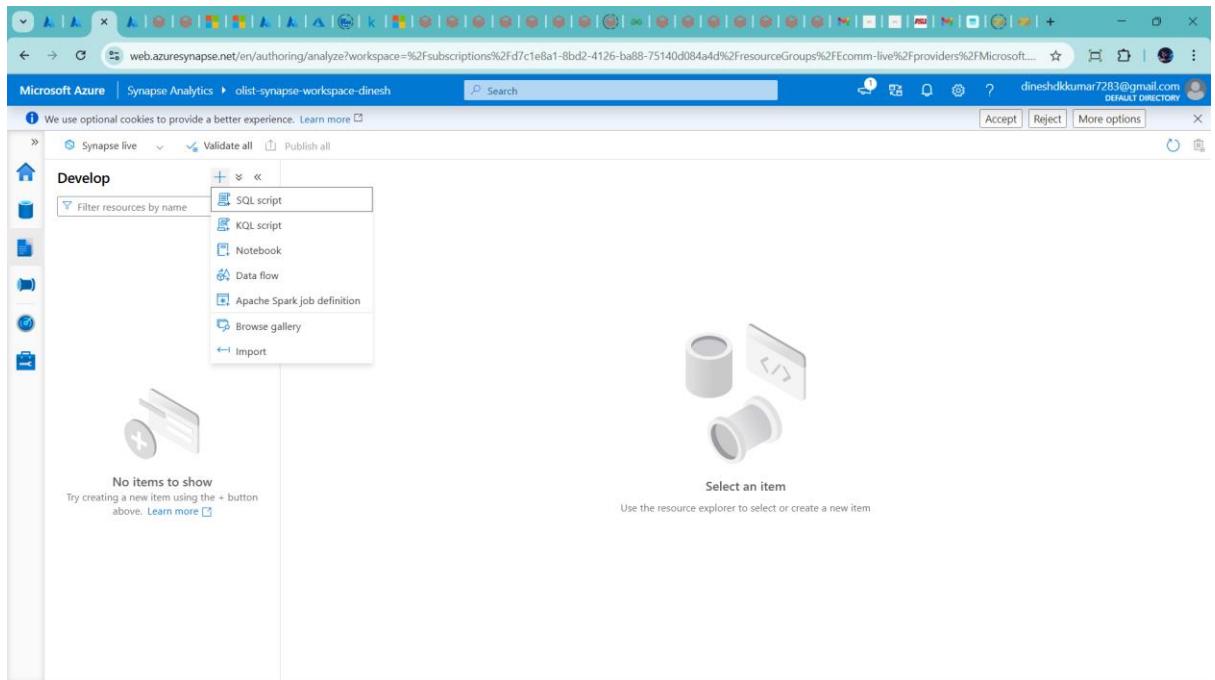
Loading

→ it is synapse

The screenshot shows the Microsoft Azure Synapse Analytics workspace home page. At the top, there's a navigation bar with the URL web.azuresynapse.net/en/home?workspace=%2Fsubscriptions%2Fd7c1e8a1-8bd2-4126-ba88-75140d084a4d%2FresourceGroups%2Fecomm-live%2Fproviders%2FMicrosoft.Synapse%2F.... Below the URL is a cookie consent message: "We use optional cookies to provide a better experience. Learn more". On the right, there are buttons for "Accept", "Reject", and "More options". The main title is "Synapse Analytics workspace olist-synapse-workspace-dinesh". A large circular graphic on the right features a bar chart and network nodes. Below the title, there are three main buttons: "Ingest" (Perform a one-time or scheduled data load.), "Explore and analyze" (Learn how to get insights from your data.), and "Visualize" (Build interactive reports with Power BI capabilities.). Further down, there are sections for "Discover more" (Knowledge center, Browse partners) and "Recent resources".

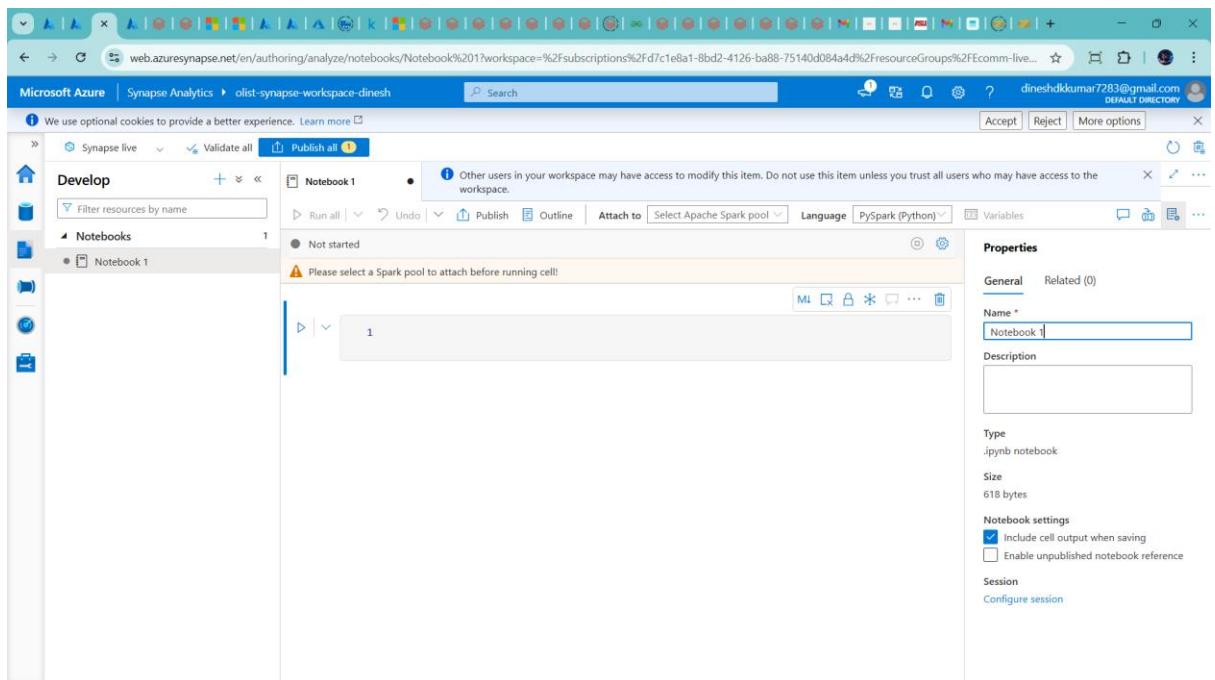
The screenshot shows the Microsoft Azure Synapse Analytics workspace Data Explorer. The left sidebar has a "Data" section with "Synapse live", "Validate all", and "Publish all" buttons. Under "Data", there are tabs for "Workspace" (selected), "Link", and "Linked". The "Workspace" tab shows a list of resources: "SQL database" (selected), "Lake database", "Data Explorer database (preview)", "Connect to external data", "Integration dataset", and "Browse gallery". To the right, there's a placeholder area with two cylinders and the text "Select an item" and "Use the resource explorer to select or create a new item".

→lakedatabse → it is create by spark



→develop →sql script,notebook

→click notebook



→click manage pool

Microsoft Azure | Synapse Analytics > olist-synapse-workspace-dinesh

Develop Notebooks Notebook 1

Notebook 1 • Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Not started Manage pools

Please select a Spark pool to attach before running cell!

Properties

General Related (0)

Name * Notebook 1

Description

Type .ipynb notebook

Size 618 bytes

Notebook settings

Include cell output when saving

Enable unpublished notebook reference

Session

Configure session

Microsoft Azure | Synapse Analytics > olist-synapse-workspace-dinesh

Analytics pools SQL pools Apache Spark pools Data Explorer pools (prev...) External connections Linked services Microsoft Purview Integration Triggers Integration runtimes Security Access control Credentials Managed private endpoints Configurations + libraries Workspace packages Data flow libraries Apache Spark configurations Source control

Apache Spark pool

Apache Spark pools can be tuned to run different kinds of Apache Spark workloads using specific configuration libraries, permissions, etc. [Learn more](#)

+ New Refresh

Filter by name

Showing 0-0 of 0 item

Name	Node size family	Size
No items to show		

Try changing your filter or create new Apache Spark pool

New Apache Spark pool

→ click new apache spark pool

→ explore concept only and cannot create anything

→ go back and delete notebook

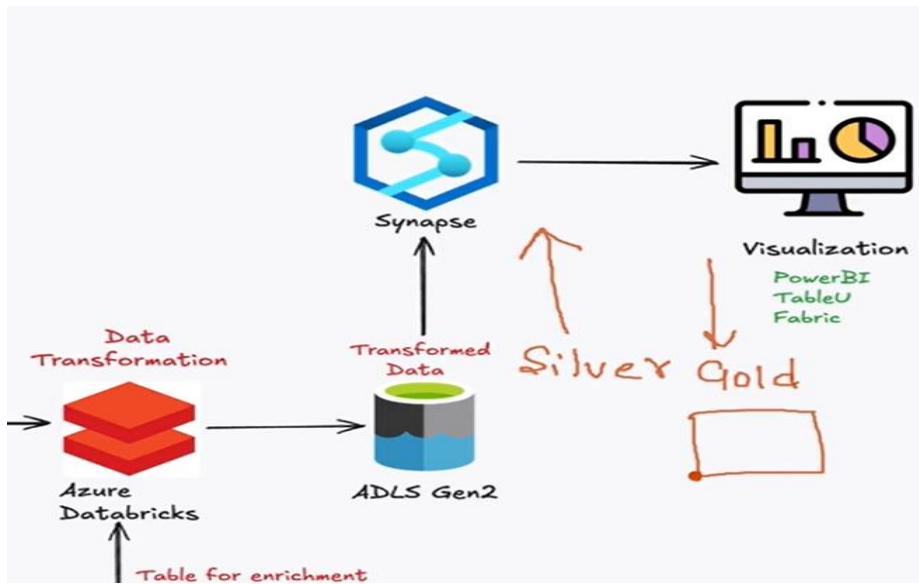
The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. In the center, there's a list of notebooks under the 'Develop' tab. One notebook, 'Notebook 1', is selected. A context menu is open over this notebook, with the 'Delete' option highlighted. To the right, a 'Properties' panel displays the following details for 'Notebook 1':

- Name: Notebook 1
- Description: (empty)
- Type: .ipynb notebook
- Size: 618 bytes
- Notebook settings:
 - Include cell output when saving
 - Enable unpublished notebook reference
- Session:
Configure session

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The 'Notebooks' section is now empty, displaying a message: 'No items to show' and 'Try creating a new item using the + button above.' On the right side, a message box appears with the following content:

Notebook 1
Notebook 1 has been deleted.

→ build gold layer using synapse



→ validate

→ go TO synapsetostorage container

The screenshot shows the Microsoft Azure Storage Explorer interface for the **synapsetostorage** container:

- Overview**: Shows basic information about the storage account.
- Activity log**: Shows recent activity.
- Tags**: Allows adding tags to the container.
- Diagnose and solve problems**: Provides troubleshooting tools.
- Access Control (IAM)**: Manages roles and permissions.
- Data migration**: Migrates data between storage accounts.
- Events**: Monitors storage events.
- Storage browser**: Provides a visual representation of data in the container.
- Partner solutions**: Offers third-party integration.
- Resource visualizer**: Visualizes storage resources.
- Data storage** (selected):
 - Containers** (selected): Shows a list of containers. One container, **synapsefilesys**, is listed with the following details:

Name	Last modified	Anonymous access level	Lease state
synapsefilesys	3/19/2025, 3:43:44 PM	Private	Available
 - File shares**
 - Queues**
 - Tables**
- Security + networking**
- Data management**
- Settings**
- Monitoring**

Click **synapsefilesys** → click upload → add customer.csv file

The screenshot shows the Microsoft Azure Storage Container Overview page for the 'synapsefilesys' container. On the right, there is an 'Upload blob' interface with a cloud icon and a 'Drag and drop files here' area, along with a 'Browse for files' link. Below this are options for 'Overwrite if files already exist' and an 'Advanced' section. At the bottom are 'Upload' and 'Give feedback' buttons.

The screenshot shows the Microsoft Azure Storage Container Overview page after a file has been uploaded. A success message at the top right says 'Successfully uploaded blob(s)' and 'Successfully uploaded 1 blob(s)'. The table below lists the uploaded file 'customers.csv' with details: Name: customers.csv, Modified: 3/19/2025, 4:01:02 PM, Access tier: Hot (Inferred), Archive status: Not yet archived, Blob type: Block blob, Size: 270.19 KB, Lease state: Available. There is also a 'Show deleted objects' toggle switch.

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
customers.csv	3/19/2025, 4:01:02 PM	Hot (Inferred)	Not yet archived	Block blob	270.19 KB	Available

→go to check auzre synapse side→click data →click Linked

The screenshot shows the Microsoft Azure Synapse Analytics Data Explorer interface. The left sidebar has a 'Data' section with 'Workspace' and 'Linked' tabs. Under 'Linked', there is a tree view of 'Azure Data Lake Storage Gen2' with a node 'olist-synapse-workspace-din...' expanded, showing 'synapsefilessys (Primary)' and '(Attached Containers)'. The main pane displays two cylinders with code snippets, with the text 'Select an item' and 'Use the resource explorer to select or create a new item'.

→click synapsefilessys

The screenshot shows the Microsoft Azure Synapse Analytics Data Explorer interface. The left sidebar has a 'Data' section with 'Workspace' and 'Linked' tabs. Under 'Linked', there is a tree view of 'Azure Data Lake Storage Gen2' with a node 'olist-synapse-workspace-din...' expanded, showing 'synapsefilessys (Primary)' and '(Attached Containers)'. The main pane shows a file list for 'synapsefilessys' with one item: 'customers.csv' (Last Modified: 3/19/2025, 4:01:02 PM, Content Type: CSV, Size: 270.2 KB). A warning message at the top right says: 'Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.'

→click new SQL script →arrow

The screenshot shows the Microsoft Azure Synapse Analytics Data blade. On the left, there's a navigation pane with 'Data' selected, showing 'Workspace' and 'Linked'. Under 'Linked', there's a section for 'Azure Data Lake Storage Gen2' with one item: 'olist-synapse-workspace-dinesh (P... synapsefilesys (Primary)'. The main area is titled 'synapsefilesys' and contains a table with the following columns: 'New SQL script', 'New notebook', 'New data flow', 'New integration dataset', 'Upload', 'Download', '+ New folder', 'Select all', and 'More'. The table has a single row: 'Select TOP 100 rows' under 'New SQL script'. Below the table, there are buttons for 'Create external table' and 'Bulk load'. A warning message at the top right says: 'Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.' At the bottom, it says 'Showing 1 to 1 of 1 cached items'.

→Top 100 rows

The screenshot shows the Microsoft Azure Synapse Analytics SQL script editor. The left sidebar shows 'Data' and 'Linked' sections. The main area has a tab for 'SQL script 1'. The code editor contains the following T-SQL script:

```
1 -- This is auto-generated code
2 SELECT
3   |TOP 100 *
4   FROM
5     OPENROWSET(
6       BULK 'https://synapselfilesys.dfs.core.windows.net/synapsefilesys/customers.csv',
7       FORMAT = 'CSV',
8       PARSE_VERSION = '2.0'
9     ) AS [result]
```

To the right of the code editor is a 'Properties' panel with tabs for 'General' and 'Related (0)'. Under 'General', there are fields for 'Name' (set to 'SQL script 1') and 'Description'. Below these are settings for 'Type' (.sql script), 'Size' (243 bytes), and 'Results settings per query' (with 'First 5000 rows (default)' checked). There are also options for 'All rows' and 'Run'.

→run

Microsoft Azure | Synapse Analytics > olist-synapse-workspace-dinesh

We use optional cookies to provide a better experience. Learn more ▾

Accept | Reject | More options

Data Workspace Linked

Filter resources by name

Azure Data Lake Storage Gen2 (2)

- olist-synapse-workspace-dinesh (P...)
- synapselfilesys (Primary)
- (Attached Containers)

SQL script 1

1 -- This is auto-generated code
2 SELECT
3 TOP 100 *
4 FROM
5 OPENROWSET(
6 BULK 'https://synapselfilesys.dfs.core.windows.net/synapselfilesys/customers.csv',
7 FORMAT = 'CSV',
8 PARSE_VERSION = '2.0'
9) AS [result]

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 243 bytes

Results settings per query ▾

First 5000 rows (default) All rows

Results

View Table Chart Export results ▾

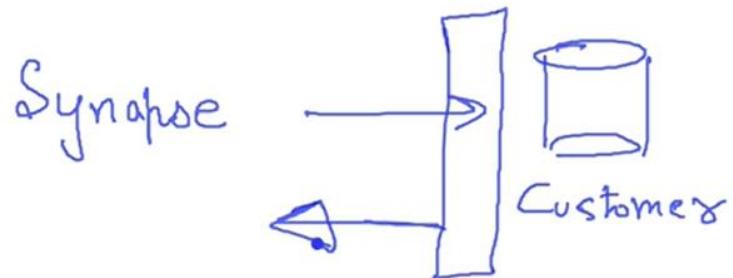
Search

C1	C2	C3	C4
customer_id	name	email	country
8d96de14	Customer_4082	customer_4082@example.com	India
2f794a68	Customer_5948	customer_5948@example.com	UK
97f47166	Customer_9793	customer_9793@example.com	Canada
0fbface3	Customer_8988	customer_8988@example.com	UK

00:00:09 Query executed successfully.

→ openrowset → it is retrieve data from ADLS Gen 2 . but they cannot store data in synapse. It data present in ADLS gen 2 storage only they share only meta data

) AS [result]



-→ Now I want to read a data from olistdatastoragedinesh

Azure Synapse

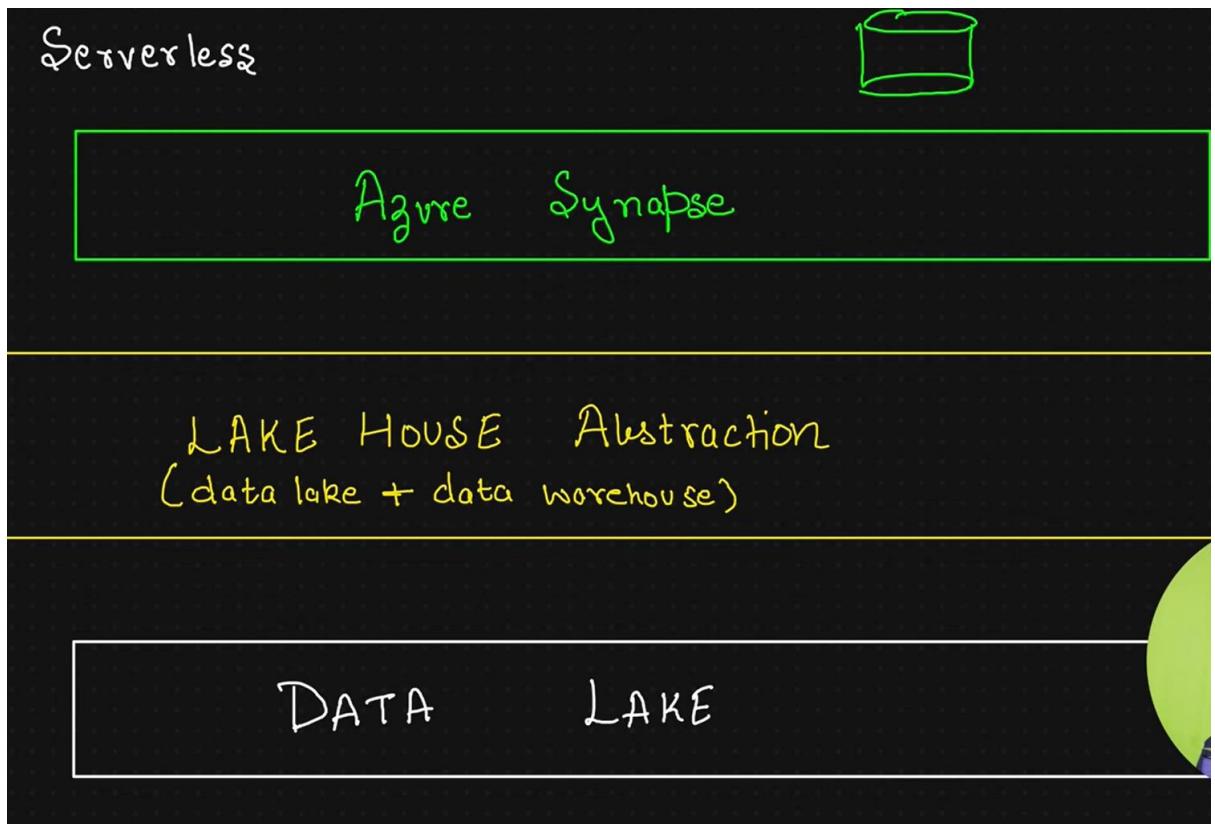


Azure
Synapse
Analytics

Azure Synapse is a cloud based data warehouse and analytics service that combines.

- data integration
- enterprise data warehousing
- Big Data analytics

To access the data in azure Synapse , we need to give permission to Synapse via ADLS gen2 .



→ data present in data lake

→ go to storage container → give permission to get data from synapse

→ click olistdatastoragedinesh → click Add role assignment

The screenshot shows the Microsoft Azure Access Control (IAM) interface. On the left, there's a navigation sidebar with various options like Overview, Activity log, Tags, and Data storage. The main area is titled "Check access" and includes sections for "My access", "Check access", "Grant access to this resource", "View access to this resource", and "View deny assignments". A "New! Permissions Management" section is also present.

→click blob container

The screenshot shows the "Add role assignment" page for a blob container. It has tabs for Role, Members*, Conditions, and Review + assign. The Role tab is selected, showing a search bar with "blob" and a table of available roles. The table includes columns for Name, Description, Type, Category, and Details. The "Storage Blob Data Contributor" role is highlighted.

Name	Description	Type	Category	Details
Defender CSPM Storage Data Scanner	Grants access to read blobs and files. This role is used by the data scanner of Defender CSPM.	BuiltinRole	None	View
Defender for Storage Data Scanner	Grants access to read blobs and update index tags. This role is used by the data scanner of Defender for Storage.	BuiltinRole	None	View
Storage Blob Data Contributor	Allows for read, write and delete access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Data Owner	Allows for full access to Azure Storage blob containers and data, including assigning POSIX access control.	BuiltinRole	Storage	View
Storage Blob Data Reader	Allows for read access to Azure Storage blob containers and data	BuiltinRole	Storage	View
Storage Blob Delegator	Allows for generation of a user delegation key which can be used to sign SAS tokens	BuiltinRole	Storage	View

Select managed identities

Subscription * Azure for Students

Managed identity Synapse workspace (1)

Search by name

Selected members:
olist-synapse-workspace-dinesh
/subscriptions/d7c1e8a1-8bd2-4126-ba88-75140d084a4d/resourceGroup... Remove

Review + assign Previous Next Select Close Feedback

→ you can muple user want to access try this below steps

→ click user, group or service principal

→ click select member

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
olist-synapse-workspace-dinesh	17b0ca49-c1bc-43e3-bade-c408cde5c6...	Synapse workspace

Description

Review + assign **Previous** **Next** **Select** **Close**

→click select

→click review +assign

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
olist-synapse-workspace-dinesh	17b0ca49-c1bc-43e3-bade-c408cde5c6...	Synapse workspace
Dinesh km(Guest)	0522c242-7ab8-4c4e-b619-2d40754071...	User

Description

Review + assign **Previous** **Next** **Feedback**

olistdatastoragedinedesh | Access Control (IAM)

My access

Check access

Grant access to this resource

View access to this resource

View deny assignments

New! Permissions Management

Syntax

`OPENROWSET` syntax is used to query external data sources:

syntaxsql

```
OPENROWSET
( 'provider_name'
, { 'datasource' ; 'user_id' ; 'password' | 'provider_string' }
, { [ catalog. ] [ schema. ] object | 'query' }
)
```

`OPENROWSET(BULK)` syntax is used to read external files:

SQL Pool :- Serverless vs Dedicated

Feature	Serverless SQL Pool (Pay-as-you-go)	Dedicated SQL Pool (Provisioned)
How it works?	You don't set up or manage servers. It is on-demand . Synapse spins up resources when you need them.	You reserve fixed resources (compute power) for your data, which are always available.
Payment Model	Pay only for the queries you run.	Pay for the reserved compute (fixed cost).
Use Case	Good for exploring large datasets or running queries occasionally.	Best for big workloads that need fast performance and predictable speed.
Data Location	Data stays in the Data Lake (e.g., ADLS Gen2).	Data is loaded into a dedicated SQL database in Synapse.
Setup Complexity	Very easy to set up—no servers to manage.	More setup needed, but provides high performance .

Bus

own Car

→ go to data → click Sql databases

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, there's a navigation sidebar with options like 'Synapse live', 'Workspace', and 'Linked'. The main area is titled 'Data' and shows a list of resources under 'Azure Data Lake Storage Gen2': 'olist-synapse-workspace-dinesh (P...)' and '(Attached Containers)'. To the right, a modal dialog is open for 'Create SQL database'. The dialog has a title 'Create SQL database' and a sub-instruction 'Create database to organize your workload into databases and database objects.' It includes a 'Select SQL pool type' section with two radio buttons: 'Serverless' (selected) and 'Dedicated'. Below that is a 'Database' input field and 'Create' and 'Cancel' buttons at the bottom.

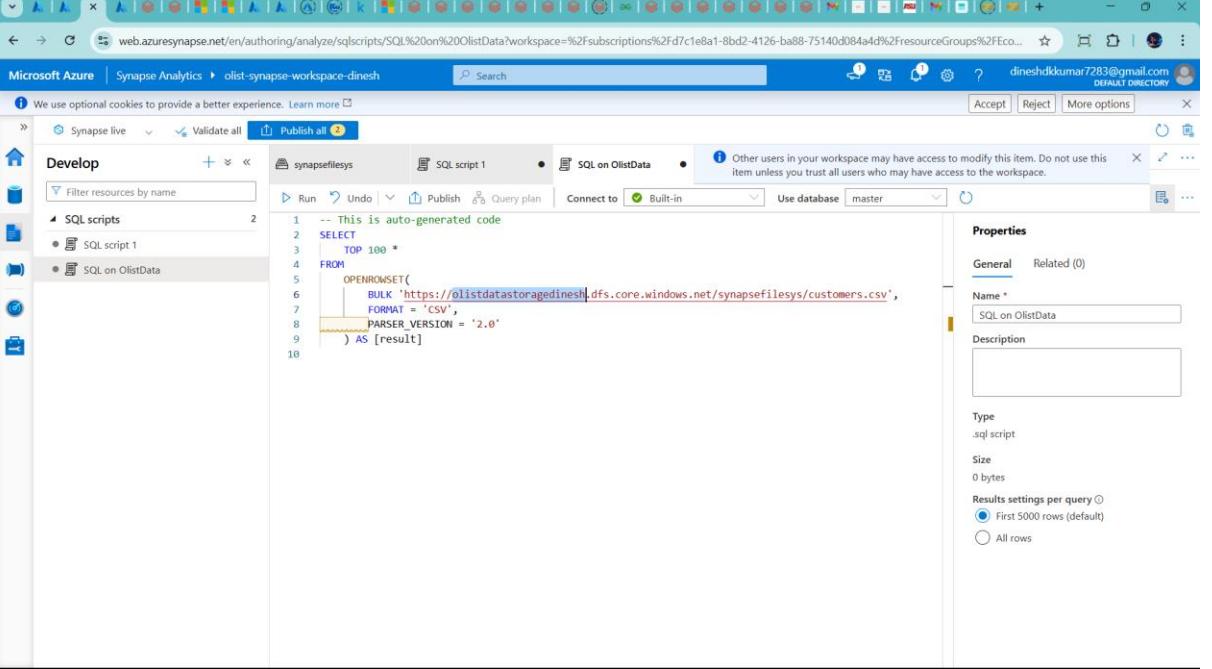
The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. On the left, there's a navigation sidebar with 'Data' selected, showing 'Workspace' and 'Linked' sections. Under 'Linked', it lists 'Azure Data Lake Storage Gen2' and 'olist-synapse-workspace-dinesh (P...)'. The main area shows two SQL scripts: 'synapsesfilesys' and 'SQL script 1'. A 'Create SQL database' dialog is open on the right, prompting for a database name 'olist'. The 'Serverless' option is selected under 'Select SQL pool type'. At the bottom right of the dialog are 'Create' and 'Cancel' buttons.

→create

→copy that name

The screenshot shows the Microsoft Azure portal's 'Access Control (IAM)' page for the storage account 'olistdatastoragedinesh'. The left sidebar has 'Access Control (IAM)' selected. The main area shows tabs for 'Check access', 'Role assignments', 'Roles', 'Deny assignments', and 'Classic administrators'. Under 'Check access', there's a 'View my access' button. Below it, there are three cards: 'Grant access to this resource', 'View access to this resource', and 'View deny assignments'. A 'New! Permissions Management' box at the bottom encourages users to discover and remediate unused permissions. The top right of the page shows the user's email 'dineshdkumar7283@gmail.com' and a 'DEFAULT DIRECTORY' button.

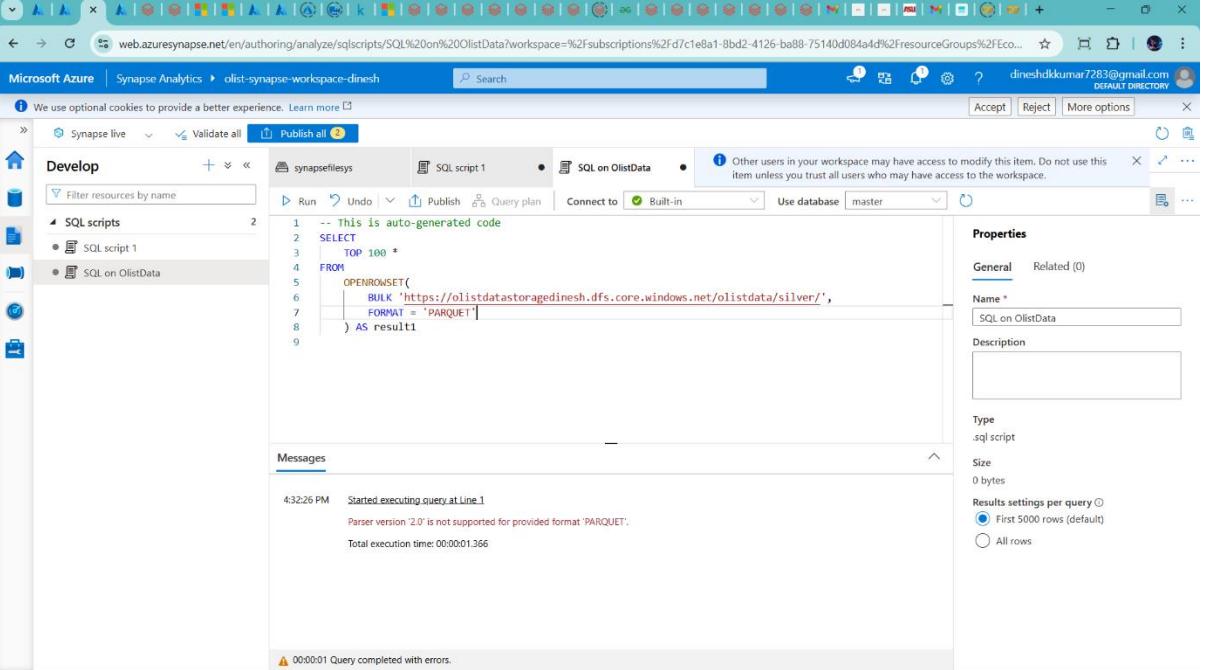
→change it



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar has a 'Develop' section with 'SQL scripts' selected. The main area displays a SQL script titled 'synapsefilesys' under 'SQL script 1'. The script reads data from a CSV file located at 'https://olistdatastorage.dfs.core.windows.net/synapsefilesys/customers.csv' using an OPENROWSET BULK command. The properties pane on the right shows the script is named 'SQL on OlistData' and is a .sql script.

```
-- This is auto-generated code
1 SELECT
2     TOP 100 *
3 FROM
4     OPENROWSET(
5         BULK 'https://olistdatastorage.dfs.core.windows.net/synapsefilesys/customers.csv',
6         FORMAT = 'CSV',
7         PARSE_VERSION = '2.0'
8     ) AS [result]
```

→



This screenshot shows the same workspace after changes. The 'FORMAT' clause in the script has been changed to 'PARQUET'. The 'Messages' pane at the bottom shows the query started executing at 4:32:26 PM, but it failed because 'Parser version '2.0'' is not supported for the provided format 'PARQUET'. The total execution time was 0:00:01.366. A warning message at the bottom indicates the query completed with errors.

```
-- This is auto-generated code
1 SELECT
2     TOP 100 *
3 FROM
4     OPENROWSET(
5         BULK 'https://olistdatastorage.dineshkumar7283@gmail.com/olistdata/silver/',
6         FORMAT = 'PARQUET'
7     ) AS result
```

→click run

```

-- This is auto-generated code
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://olistdatastoragedinedesh.dfs.core.windows.net/olistdata/silver/',
        FORMAT = 'PARQUET'
    ) AS result

```

product_category	order_status	order_purchase_timestamp	order_approved_timestamp	order_delivered_carrier_timestamp	order_delivered_customer_timestamp	order_estimated_delivery_timestamp
eletronics	delivered	2018-02-10T17...	2018-02-10T18...	2018-02-19T21...	2018-02-20T22...	2018-02-26T01...
pet_shop	delivered	2017-09-21T22...	2017-09-21T22...	2017-09-22T17...	2017-09-26T19...	2017-10-13T01...
fashion_bolsas...	delivered	2017-01-16T14...	2017-01-16T14...	2017-01-16T15...	2017-01-23T08...	2017-02-24T01...
eletrodomesticos	delivered	2018-01-30T11...	2018-01-30T11...	2018-02-05T15...	2018-02-21T17...	2018-03-13T01...
cama_mesa_ban...	delivered	2018-08-16T12...	2018-08-16T13...	2018-08-17T15...	2018-08-20T20...	2018-08-21T01...
beleza_saude	delivered	2018-01-17T12...	2018-01-18T02...	2018-01-19T00...	2018-01-25T17...	2018-02-08T01...

00:00:06 Query executed successfully.

→ path name already present in storage container

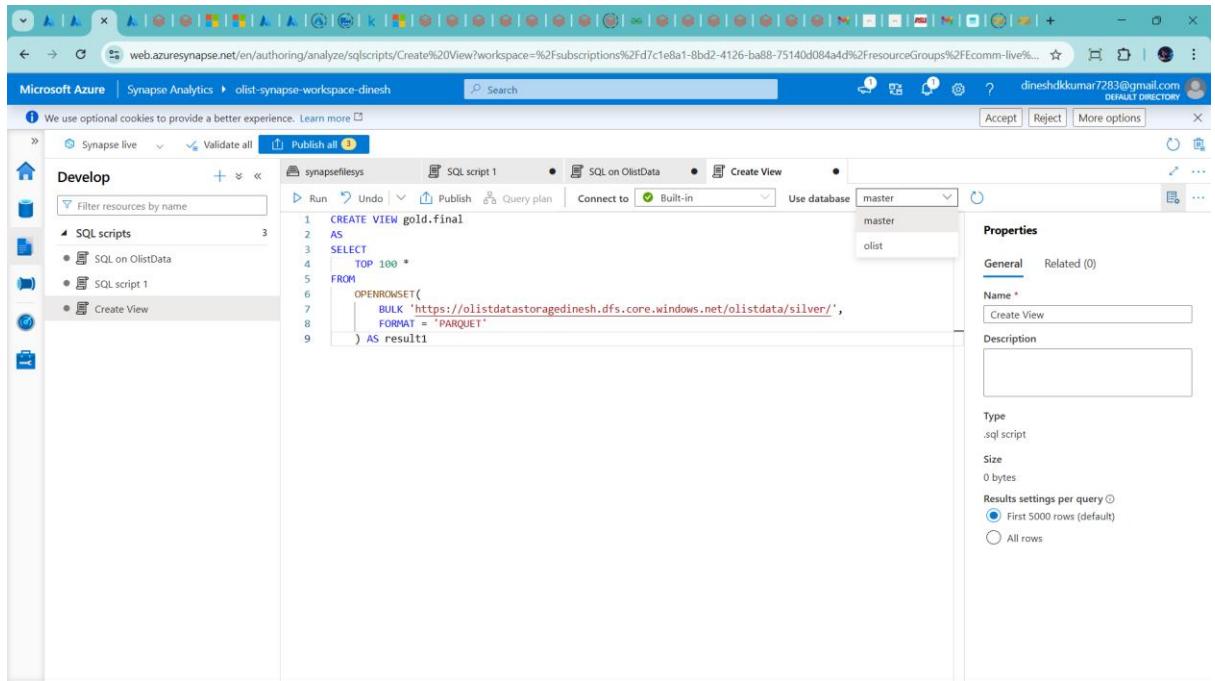
silver/part-00003-tid-7333565244183789893-2a347ec1-7314-4609-941f-e8...

Name	...
[-]	...
_committed_73335652441837...	...
_started_7333565244183789893	...
_SUCCESS	...
part-00000-tid-733356524418...	...
part-00001-tid-733356524418...	...
part-00002-tid-733356524418...	...
part-00003-tid-733356524418...	...

Properties

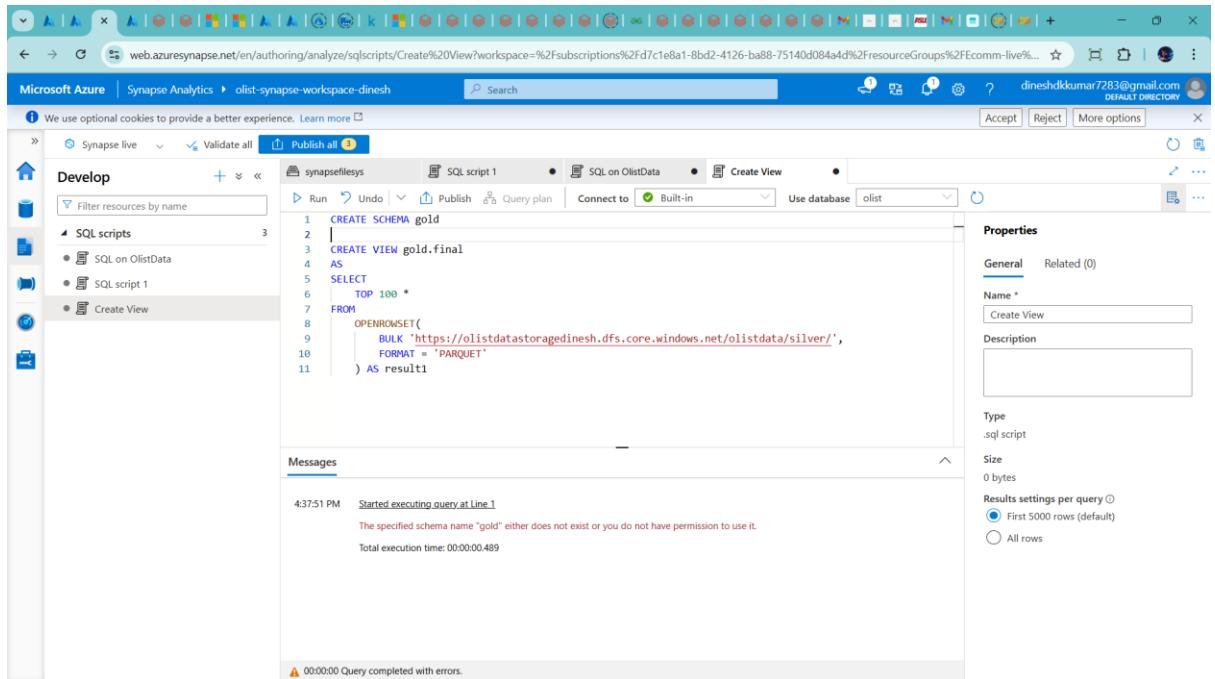
URL	https://olistdatastorage...
LAST MODIFIED	3/19/2025, 3:08:38 PM
CREATION TIME	3/19/2025, 3:08:38 PM
VERSION ID	-
TYPE	Block blob
SIZE	3.1 MiB
ACCESS TIER	Hot (Inferred)
ACCESS TIER LAST MODIFIED	N/A
ARCHIVE STATUS	-
REHYDRATE PRIORITY	-
SERVER ENCRYPTED	true
ETAG	0x8DD66C9D41702F1
VERSION-LEVEL IMMUTABILITY POLICY	Disabled
CACHE-CONTROL	
CONTENT-TYPE	application/octet-stream
CONTENT-MD5	
CONTENT-ENCODING	
CONTENT-LANGUAGE	
CONTENT-DISPOSITION	

→ change database master to olist



```
CREATE VIEW gold.final
AS
SELECT
TOP 100 *
FROM
OPENROWSET(
    BULK 'https://olistdatastorage.dineshkumar7283@gmail.com/olistdata/silver/',
    FORMAT = 'PARQUET'
) AS result1
```

and run → successful create view



```
CREATE SCHEMA gold
CREATE VIEW gold.final
AS
SELECT
TOP 100 *
FROM
OPENROWSET(
    BULK 'https://olistdatastorage.dineshkumar7283@gmail.com/olistdata/silver/',
    FORMAT = 'PARQUET'
) AS result1
```

Messages

4:37:51 PM Started executing query at Line 1
The specified schema name "gold" either does not exist or you do not have permission to use it.
Total execution time: 0:00:00.489

00:00:00 Query completed with errors.

```

CREATE SCHEMA gold
CREATE VIEW gold.final
AS
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://olistdatastorage.datalake.azure.net/olistdata/silver/',
        FORMAT = 'PARQUET'
    ) AS result

```

No results to show
Your query yielded no displayable results

00:00:00 Query executed successfully.

product_category	order_status	order_purchase_date	order_approved_at	order_delivered_at	order_delivered_carrier_time	order_estimated_delivery_time	actual_delivery_time	estimated_delivery_time	Delay
electronics	delivered	2018-02-10T17...	2018-02-10T18...	2018-02-19T21...	2018-02-20T12...	2018-02-26T00...	10	16	-6
pet_shop	delivered	2017-09-21T22...	2017-09-21T22...	2017-09-22T17...	2017-09-26T19...	2017-10-13T00...	5	22	-17
fashion_bolsas...	delivered	2017-01-16T14...	2017-01-16T14...	2017-01-16T15...	2017-01-23T08...	2017-02-24T00...	7	39	-32
eletrodomesticos	delivered	2018-01-30T11...	2018-01-30T11...	2018-02-05T15...	2018-02-21T17...	2018-03-13T00...	22	42	-20

00:00:01 Query executed successfully.

→create another view

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar has a 'Develop' section with 'SQL scripts' selected, showing options like 'Create View', 'SQL on OlistData', 'SQL script 1', and 'View Final'. The main area contains a code editor with the following SQL script:

```
1 2  CREATE VIEW gold.final1|  
3  AS  
4  SELECT  
5    TOP 100 *  
6  FROM  
7    OPENROWSET(  
8      BULK 'https://olistdatastorage.dinesh.core.windows.net/olistdata/silver/',  
9      FORMAT = 'PARQUET'  
10     ) AS result1  
11   WHERE order_status='delivered'
```

The 'Properties' pane on the right shows the view is named 'gold.final1' and is a '.sql script'. The 'Results' tab is currently inactive.

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface after the query has been executed. The 'Results' tab is now active, displaying a magnifying glass icon and the message 'No results to show'. Below this, it says 'Your query yielded no displayable results'. The 'Messages' tab shows a log entry: '4:44:45 PM Started executing query at Line 1' and '00:00:01 Executing query.' The 'Properties' pane on the right remains the same as in the previous screenshot.

Azure Synapse Workflow

0. Medallin architecture and Lakehouse (Pre-requisite)
1. Create a serverless SQL pool. (for just reading/Querying)
2. Create Schema & then User / password
3. Use OPENROWSET to read data from silver layer.
(here data is not stored, just referenced to silver layer)
4. Create a view to easily Query the data.

Filter by title

CETAS

CTAS

Data types

Distributed tables

Table constraints

External tables

Indexes

Identity

OPENROWSET

Partitions

Replicated tables

Statistics

Temporary

> T-SQL language elements

> Querying

> Monitoring

> Security

> Workload management

> How-to guides

> Data Explorer

> Apache Spark

> Synapse Link

> Pipeline and data flow

> Machine Learning

> Data Catalog and Governance

Learn / Azure / Synapse Analytics / Article • 10/14/2024 • 12 contributors

CETAS with Synapse SQL

In this article

- CETAS in dedicated SQL pool
- CETAS in serverless SQL pool
- Examples
- Supported data types
- Next step

You can use `CREATE EXTERNAL TABLE AS SELECT`(CETAS) in dedicated SQL pool or serverless SQL pool to complete the following tasks:

- Create an external table
- Export, in parallel, the results of a Transact-SQL SELECT statement to:
 - Hadoop
 - Azure Storage Blob
 - Azure Data Lake Storage Gen2

CETAS in dedicated SQL pool

For dedicated SQL pool, CETAS usage and syntax, check the [CREATE EXTERNAL TABLE AS SELECT](#) article. Additionally, for guidance on CTAS using dedicated SQL pool, see the [CREATE TABLE AS SELECT](#) article.

CETAS in serverless SQL pool

OVERVIEW
CETAS
CTAS
Data types
Distributed tables
Table constraints
External tables
Indexes
Identity
OPENROWSET
Partitions
Replicated tables
Statistics
Temporary
T-SQL language elements
> Querying
> Monitoring
> Security
> Workload management
> How-to guides
Data Explorer
Apache Spark
Synapse Link
Pipeline and data flow
Machine Learning
Data Catalog and Governance
How-to
Guidance
Reference
Resources
Download PDF

```
-- you can query the newly created external table
SELECT * FROM population_by_year_state
```

General example

In this example we can see example of a template code for writing CETAS with a View as source and using Managed Identity as an authentication.

```
SQL Copy

CREATE DATABASE [<mydatabase>];
GO

USE [<mydatabase>];
GO

CREATE MASTER KEY ENCRYPTION BY PASSWORD = '<strong password>';

CREATE DATABASE SCOPED CREDENTIAL [WorkspaceIdentity] WITH IDENTITY = 'Managed Identity';
GO

CREATE EXTERNAL FILE FORMAT [ParquetFF] WITH (
    FORMAT_TYPE = PARQUET,
    DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
);
GO

CREATE EXTERNAL DATA SOURCE [SynapseSQLwriteable] WITH (
    LOCATION = 'https://<mystorageaccount>.dfs.core.windows.net/<mycontainer>/<mybaseoutputfolderpath>',
    CREDENTIAL = [WorkspaceIdentity]
);
GO

CREATE EXTERNAL TABLE [dbo].[<myexternaltable>] WITH (
    LOCATION = '<myoutputsuffix>',
    DATA_SOURCE = [SynapseSQLwriteable],
    FILE_FORMAT = [ParquetFF]
) AS
SELECT * FROM [<myview>];
GO
```

Supported data types

→ I want to create external table to store data in gold layer

→ now create master key password

→ first click new sql script → name as SQL to gold layer

```
1
2  use olist
3  drop master key
4
5
6  select * from sys.database_credentials
7  drop database scoped CREDENTIAL mayankadmin
```

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar is titled 'Develop' and lists options like 'SQL scripts', 'Create View', 'SQL on OlistData', 'SQL script 1', 'SQL to gold layer', and 'View Final'. The main area contains a SQL editor with the following script:

```
1 CREATE MASTER KEY ENCRYPTION BY PASSWORD='DIN123@kum'
2 CREATE DATABASE SCOPED CREDENTIAL dineshadmin WITH IDENTITY='Managed Identity';
```

The 'Properties' pane on the right shows the following details:

- Name: SQL to gold layer
- Type: .sql script
- Size: 0 bytes
- Results settings per query:
 - First 5000 rows (default)
 - All rows

The 'Results' pane below the editor shows the message: "No results to show Your query yielded no displayable results".

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar is titled 'Develop' and lists options like 'SQL scripts', 'Create View', 'SQL on OlistData', 'SQL script 1', 'SQL to gold layer', and 'View Final'. The main area contains a SQL editor with the following script:

```
1 CREATE MASTER KEY ENCRYPTION BY PASSWORD='DIN123@kum'
2 CREATE DATABASE SCOPED CREDENTIAL dineshadmin WITH IDENTITY='Managed Identity';
3
4 SELECT * FROM sys.database_credentials
```

The 'Properties' pane on the right shows the following details:

- Name: SQL to gold layer
- Type: .sql script
- Size: 0 bytes
- Results settings per query:
 - First 5000 rows (default)
 - All rows

The 'Results' pane below the editor shows a table with one row of data:

name	principal_id	credential_id	credential_id...	create_date	modify_date	target_type
dineshadmin	1	65536	Managed Ident...	2025-03-19T11:...	2025-03-19T11:...	(NULL)

The message "00:00:00 Query executed successfully." is displayed at the bottom.

Data Migration to Gold layer

Input



format - parquet

Source - When we read the data

Output



format → parquet
source

CETAS ⇒ Create external table as select

The screenshot shows a browser window with the URL learn.microsoft.com/en-us/azure/synapse-analytics/sql/develop-tables-cetas. The page title is "CREATE EXTERNAL TABLE AS SELECT (CETAS) - Microsoft Learn". On the left, there's a sidebar with a navigation tree under "CETAS" including "CTAS", "Data types", "Distributed tables", "Table constraints", "External tables", "Indexes", "Identity", "OPENROWSET", "Partitions", "Replicated tables", "Statistics", "Temporary", "T-SQL language elements" (with sub-items "Querying", "Monitoring", "Security", "Workload management"), "How-to guides", "Data Explorer", "Apache Spark", "Synapse Link", and "Download PDF". The main content area contains a "SQL" code editor with the following T-SQL script:

```
CREATE DATABASE [];
GO

USE [];
GO

CREATE MASTER KEY ENCRYPTION BY PASSWORD = '<strong password>';

CREATE DATABASE SCOPED CREDENTIAL [workspaceIdentity] WITH IDENTITY = 'Managed Identity';
GO

CREATE EXTERNAL FILE FORMAT [ParquetFF] WITH (
    FORMAT_TYPE = PARQUET,
    DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
);
GO

CREATE EXTERNAL DATA SOURCE [SynapseSQLwriteable] WITH (
    LOCATION = 'https://<mystorageaccount>.dfs.core.windows.net/<mycontainer>/<mybaseoutput>',
    CREDENTIAL = [workspaceIdentity]
);
GO

CREATE EXTERNAL TABLE [dbo].[<myexternaltable>] WITH (
    LOCATION = '<myoutputsuffix>',
    DATA_SOURCE = [SynapseSQLwriteable],
    FILE_FORMAT = [ParquetFF]
) AS
SELECT * FROM [<myview>];
GO
```

To the right of the code editor, there are sections for "Training" (with links to "Use Azure Synapse serverless SQL pools to transform data in a data lake - Training" and "Use Azure Synapse serverless SQL pools to transform data in a data lake"), "Certification" (link to "Microsoft Certified: Azure Data Engineer Associate - Certifications"), and "Documentation" (links to "Troubleshoot the Parquet format connector - Azure Data Factory & Azure Synapse", "Use External Tables with Synapse SQL - Azure Synapse Analytics", "Reading or writing data files with external tables in Synapse SQL", and "Troubleshoot the Azure Data Lake Storage connectors - Azure Data Factory & Azure Synapse"). A "Show 5 more" link is also present.

The screenshot shows the Microsoft Azure Synapse Analytics studio interface. In the left sidebar under 'Develop', 'SQL script 1' is selected. The main area contains the following SQL code:

```

1 --CREATE MASTER KEY ENCRYPTION BY PASSWORD='DIN123@kum'
2 --CREATE DATABASE SCOPED CREDENTIAL dineshadmin WITH IDENTITY='Managed Identity';
3
4 SELECT * from sys.database_credentials
5
6 CREATE EXTERNAL FILE FORMAT extfileformat WITH (
7     FORMAT_TYPE = PARQUET,
8     DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
9 );
10 CREATE EXTERNAL DATA SOURCE silverlayer WITH (
11     LOCATION = 'https://olistdatastorage.dinesh.dfs.core.windows.net/olistdata/gold/',
12     CREDENTIAL = dineshadmin
13 );
14 CREATE EXTERNAL TABLE [dbo].[myexternaltable] WITH (
15     LOCATION = '<myoutputsubfolders>',
16     DATA_SOURCE = [SynapseSQLwriteable],
17     FILE_FORMAT = extfileformat
18 ) AS
19 SELECT * FROM gold.final2;
20
21

```

The 'Properties' pane on the right shows the object is named 'SQL to gold layer' and is a '.sql script'. The results pane shows a table with one row:

name	principal_id	credential_id	credential_id...	create_date	modify_date	target_type
dineshadmin	1	65536	Managed Ident...	2025-03-19T11:...	2025-03-19T11:...	(NULL)

Message bar at the bottom: 00:00:00 Query executed successfully.

--run all one by one

The screenshot shows the Microsoft Azure Synapse Analytics studio interface. In the left sidebar under 'Develop', 'SQL scripts' is selected. The main area contains the same SQL code as the previous screenshot. The 'Properties' pane on the right shows the object is named 'SQL to gold layer' and is a '.sql script'. The results pane shows a message: '5:02:1 PM Started executing query at Line 14'. The message bar at the bottom: 00:00:06 Retrieving query result.

→ change mistake

Microsoft Azure | Synapse Analytics > olist-synapse-workspace-dinesh

We use optional cookies to provide a better experience. Learn more ▾

Accept | Reject | More options

Develop

Filter resources by name

- SQL scripts
- Create View
- SQL on OlistData
- SQL script 1
- SQL to gold layer**
- View Final

```

5
6 CREATE EXTERNAL FILE FORMAT extfileformat WITH (
7   FORMAT_TYPE = PARQUET,
8   DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
9 );
10 CREATE EXTERNAL DATA SOURCE goldlayer WITH (
11   LOCATION = 'https://olistdatastoragedinesh.dfs.core.windows.net/olistdata/gold/',
12   CREDENTIAL = dineshadmin
13 );
14 CREATE EXTERNAL TABLE gold.finaltable WITH (
15   LOCATION = 'Server',
16   DATA_SOURCE = goldlayer,
17   FILE_FORMAT = extfileformat
18 ) AS
19 SELECT * FROM gold.final2;
20

```

Properties

General Related (0)

Name * SQL to gold layer

Description

Type .sql script

Size 0 bytes

Results settings per query ▾

First 5000 rows (default)

All rows

Results Messages

No results to show

Your query yielded no displayable results

00:00:02 Query executed successfully.

Microsoft Azure | Synapse Analytics > olist-synapse-workspace-dinesh

We use optional cookies to provide a better experience. Learn more ▾

Accept | Reject | More options

Develop

Filter resources by name

- SQL scripts
- Create View
- SQL on OlistData
- SQL script 1
- SQL script 2**
- SQL to gold layer
- View Final

```
1 SELECT * from gold.finaltable
```

Properties

General Related (0)

Name * SQL script 2

Description

Type .sql script

Size 0 bytes

Results settings per query ▾

First 5000 rows (default)

All rows

Results Messages

View Table Chart Export results ▾

Search

product_categ...	order_status	order_purchas...	order_approve...	order_delivere...	order_delivere...	order_estimat...
eletronicos	delivered	2018-02-10T17...	2018-02-10T18...	2018-02-19T21...	2018-02-20T22...	2018-02-26T01...
pet_shop	delivered	2017-09-21T22...	2017-09-21T22...	2017-09-22T17...	2017-09-26T19...	2017-10-13T01...
fashion_bancas	delivered	2017-01-16T14...	2017-01-16T14...	2017-01-16T15...	2017-01-23T08...	2017-02-24T01...

00:00:00 Query executed successfully.

The screenshot shows the Microsoft Azure Storage Container Overview page for the 'olistdata' container. The left sidebar includes 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', and 'Settings'. The main area displays blob details with columns: Name, Modified, Access tier, Archive status, Blob type, Size, and Lease state. Two blobs are listed: one named '-' modified on 3/19/2025 at 5:11:31 PM and another named '3EEE6864-BFDD-4C07-813B-A2DE056B0D78_14_0...' modified on 3/19/2025 at 5:11:32 PM.

→publish all

The screenshot shows the Microsoft Azure Synapse Analytics workspace titled 'web.azure-synapse.net/en/authoring/analyze/sqlscripts/SQL%20script%202?workspace=%2Fsubscriptions%2Fd7c1e8a1-8bd2-4126-ba88-75140d084a4d%2FresourceGroups%2Fcomm-li...'. The left sidebar has 'Develop' selected, showing 'SQL scripts' with items like 'Create View', 'SQL on OlistData', 'SQL script 1', 'SQL script 2', 'SQL to gold layer', and 'View Final'. The main area shows a 'Publishing' tab with a 'Pending changes (6)' section. The table lists changes under 'NAME', 'CHANGE', and 'EXISTING'. Changes include 'SQL script 1' (New), 'SQL on OlistData' (New), 'Create View' (New), 'View Final' (New), 'SQL to gold layer' (New), and 'SQL script 2' (New). Below the table, a results grid shows data from the 'gold.finalltable' view, and at the bottom, a message says '000000 Query executed successfully.'

→use this to connect any Bi tools use serverless endpoint

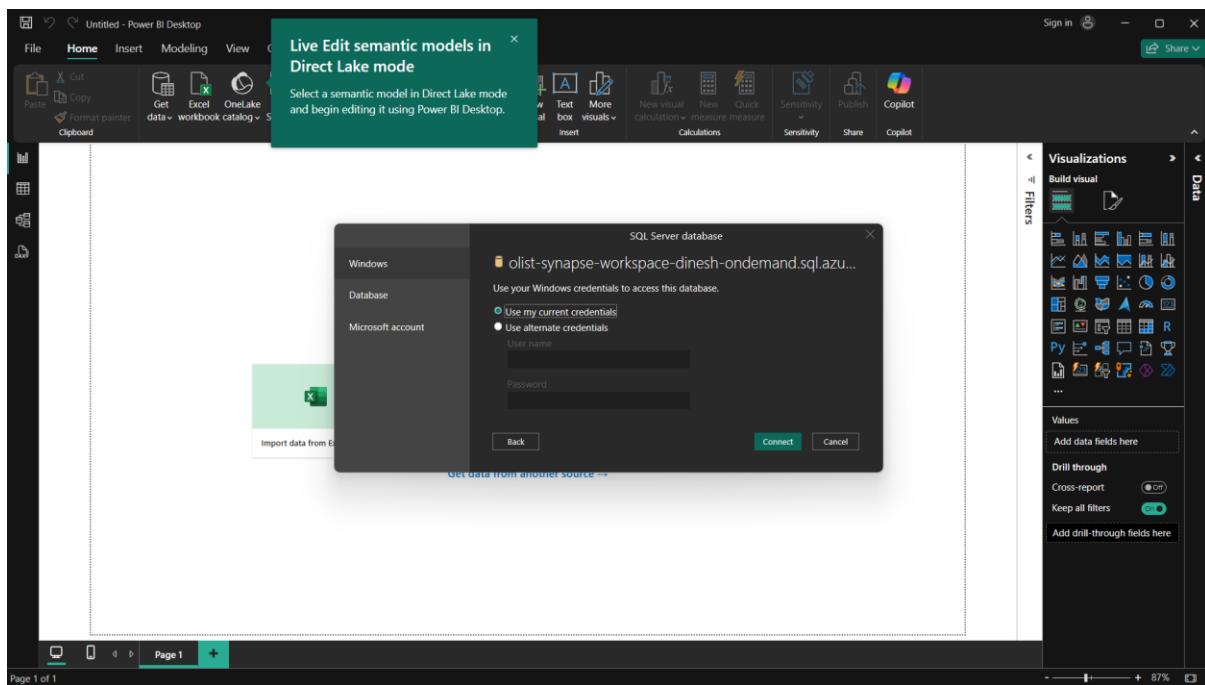
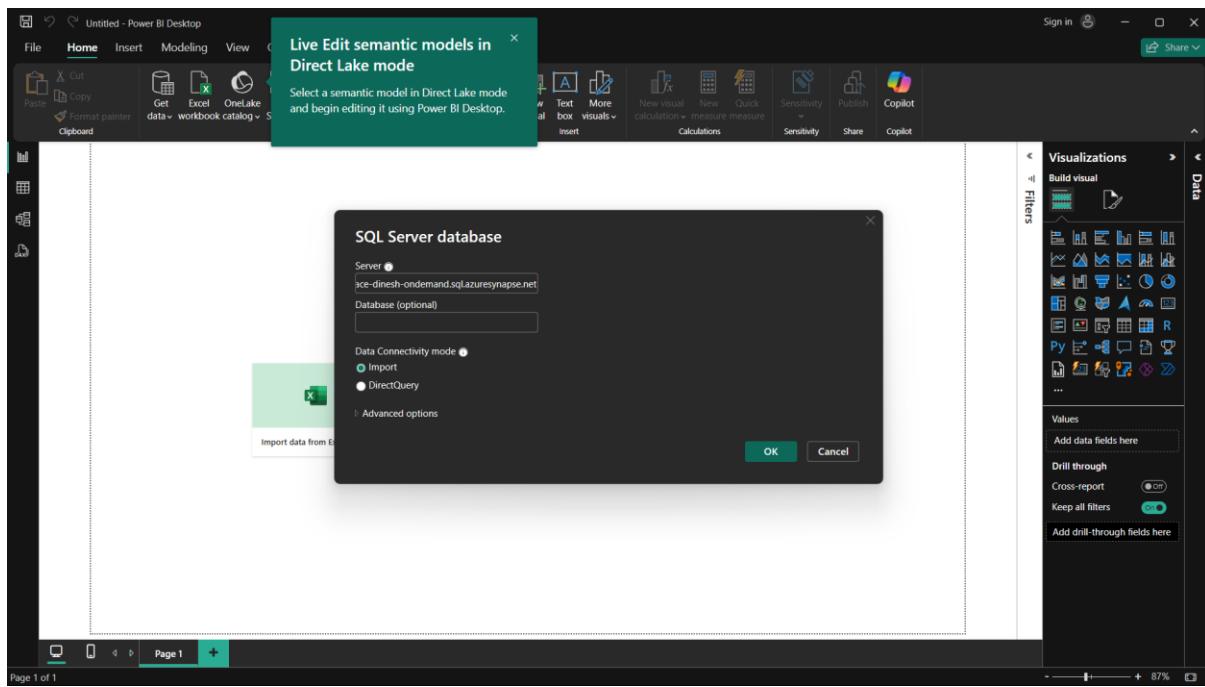
The screenshot displays two windows side-by-side. The left window is the Microsoft Azure portal showing the overview of an Ecomm-live workspace. The right window is Power BI Desktop showing the 'Get Data' dialog.

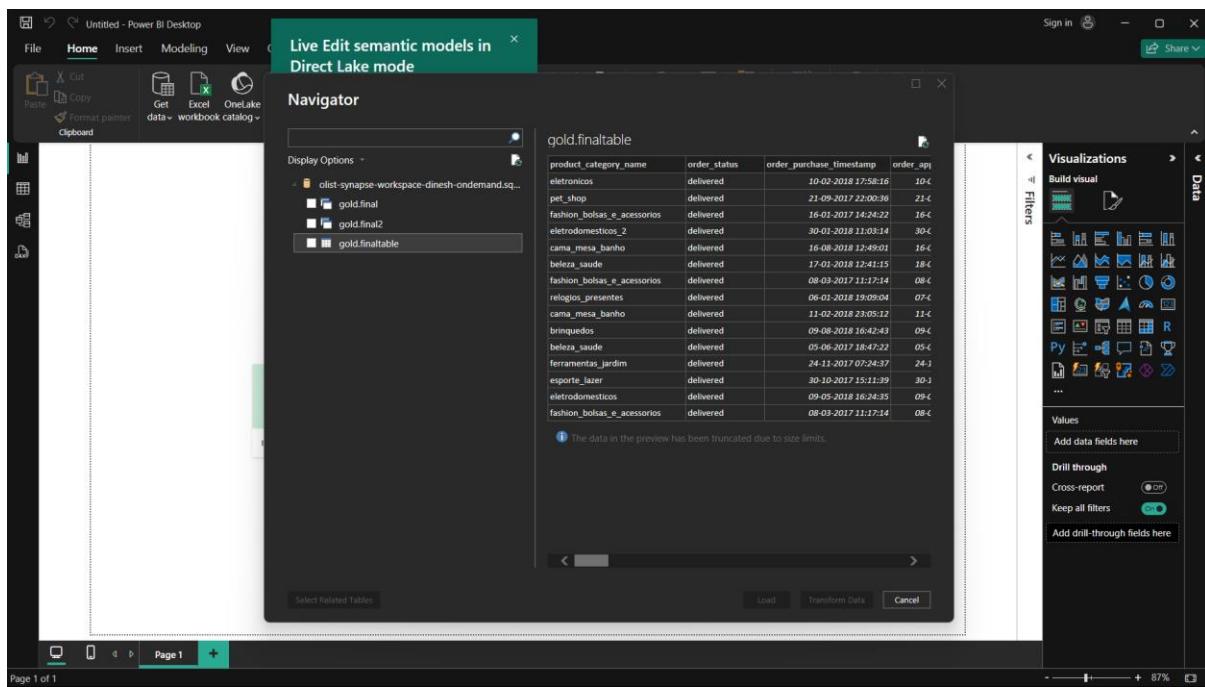
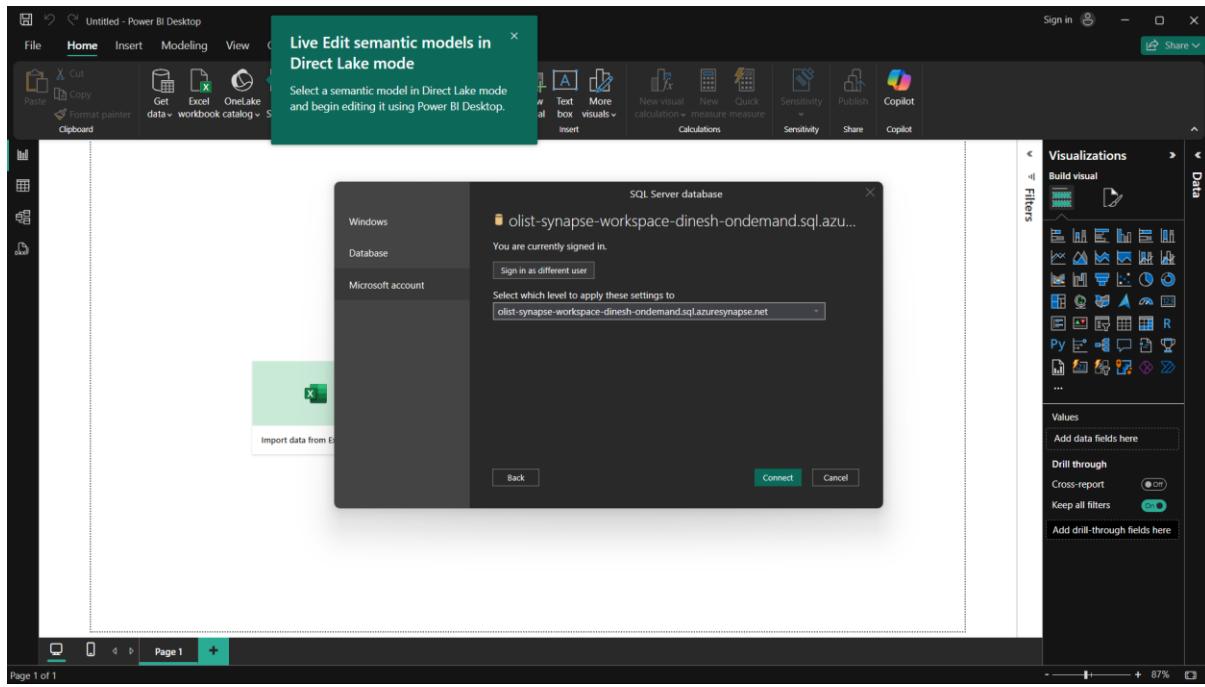
Azure Portal Overview (Left):

- Essentials:**
 - Resource group (move) : Ecomm-live
 - Status : Succeeded
 - Location : Central India
 - Subscription (move) : Azure for Students
 - Subscription ID : d7c1e8a1-8bd2-4126-ba88-75140d084a4d
 - Managed virtual network : No
 - Managed identity object id : 17b0ca49-c1bc-43e3-bade-c408cde5c652
 - Workspace web URL : <https://web.azuresynapse.net/workspace=%2bsubscriptions%2fd7c1e8a1-8bd2-4126-ba88-75140d084a4d>
 - Tags (edit) : Add tags
- Getting started:**
 - Open Synapse Studio
 - Read documentation
- Analytics pools:** A search bar and a table with columns Name, Type, and Size.

Power BI Desktop (Right):

The 'Get Data' dialog is open, showing the 'Azure Synapse Analytics SQL' connector selected from the list. The interface includes a preview area with an Excel icon, a 'Connect' button, and a 'Cancel' button. The Power BI ribbon is visible at the top, and the 'Visualizations' pane is on the right.





The screenshot shows the Power BI Desktop interface. A single visual card is displayed in the center, showing the value "112" with the subtitle "Sum of order_item_id". The "Data / Drill" tab is selected in the ribbon. The details pane on the right shows the following filters:

- Filters on this visual:
 - Sum of order_item_id is (All)
- Filters on this page:
 - Add data fields here
- Filters on all pages:
 - Add data fields here

The "Fields" section lists various data items, including:

- customer_city
- customer_state
- customer_unique_id
- customer_zip_code_prefix
- Delay Time
- estimated_delivery_time
- freight_value
- order_approved_at
- order_delivered_carrier_date
- order_estimated_customer_da...
- order_estimated_delivery_date
- order_item_id
- order_purchase_timestamp
- order_status
- payment_installments
- payment_sequential
- payment_type
- payment_value
- price
- product_category_name
- product_category_name_eng...
- product_description_lenght
- product_height_cm
- product_length_cm

Will connect with DOMO

The screenshot shows the Domo Data Center interface at <https://gwcteq-partner.domo.com/datacenter/datasources>. The search bar contains "syn" and the results show two connectors:

- Synthesio V2 Connector
- Azure Synapse SQL Connector

The interface includes a sidebar with "QUICK", "RECENT", "FAVORITES", and "..." options. The bottom status bar shows the URL, session information (Databricks PAT + Domo), user K.M.Dinesh, 1.2K views, 0 likes, and the date Mar 10, 2025 11:25 AM.

<https://gwcteq-partner.domo.com/datacenter/datasources>

CREATE NEW AZURE SYNAPSE SQL DATASET

Credentials

Azure Synapse SQL Connector

Azure Synapse Analytics lets you quickly implement a high-performance, globally available, and secure cloud data warehouse. Use Domo's Azure Synapse SQL connector (formerly the Azure SQL Data Warehouse connector) to bring your Azure data into Domo. Combine your Azure data with data from other data sources throughout your company for a comprehensive view of your business. Set up custom alerts to be notified in real-time when your key metrics change, so you can make faster, better business decisions.

LEARN MORE ABOUT THIS CONNECTOR

Azure Synapse SQL Account

JDBC DRIVER: 8.4.1

* SERVER NAME: olist-synapse-workspace-dinesh-on-demand.sql.azuresyn...

* DATABASE NAME: olist

* PORT: 1433

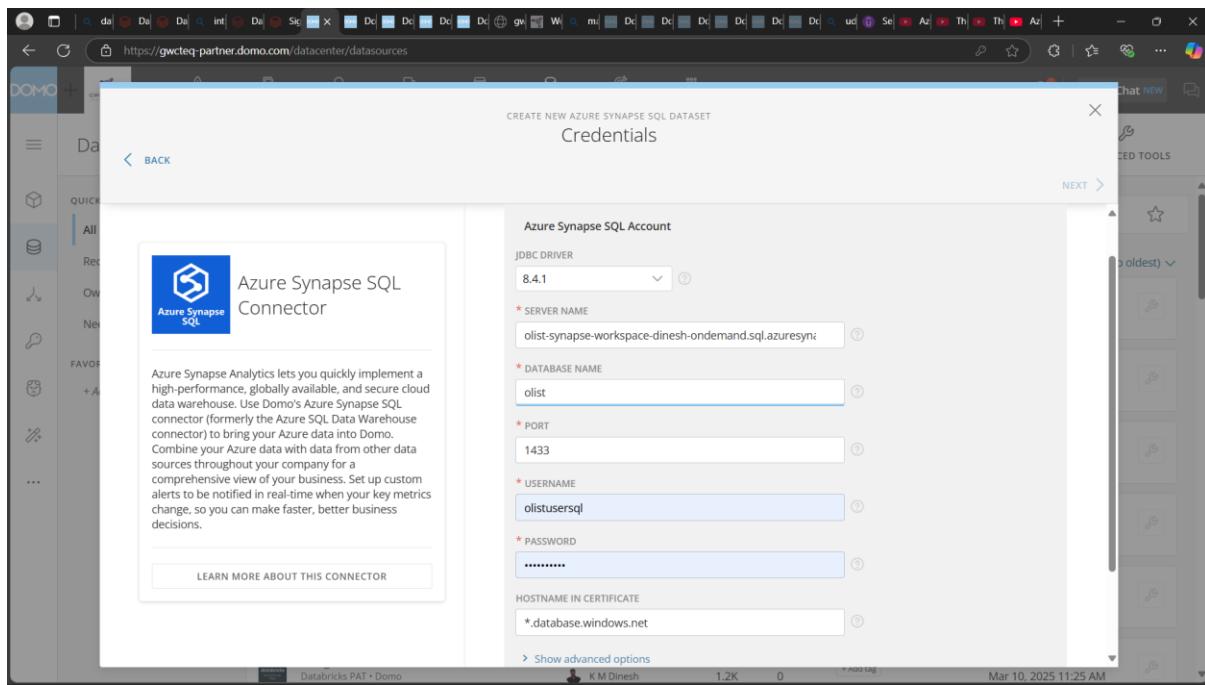
* USERNAME: olistusersql

* PASSWORD: [REDACTED]

HOSTNAME IN CERTIFICATE: *.database.windows.net

NEXT >

Databricks PAT • Domo K M Dinesh 1.2K 0 + Post tag Mar 10, 2025 11:25 AM



<https://gwcteq-partner.domo.com/datacenter/datasources>

CREATE NEW AZURE SYNAPSE SQL DATASET

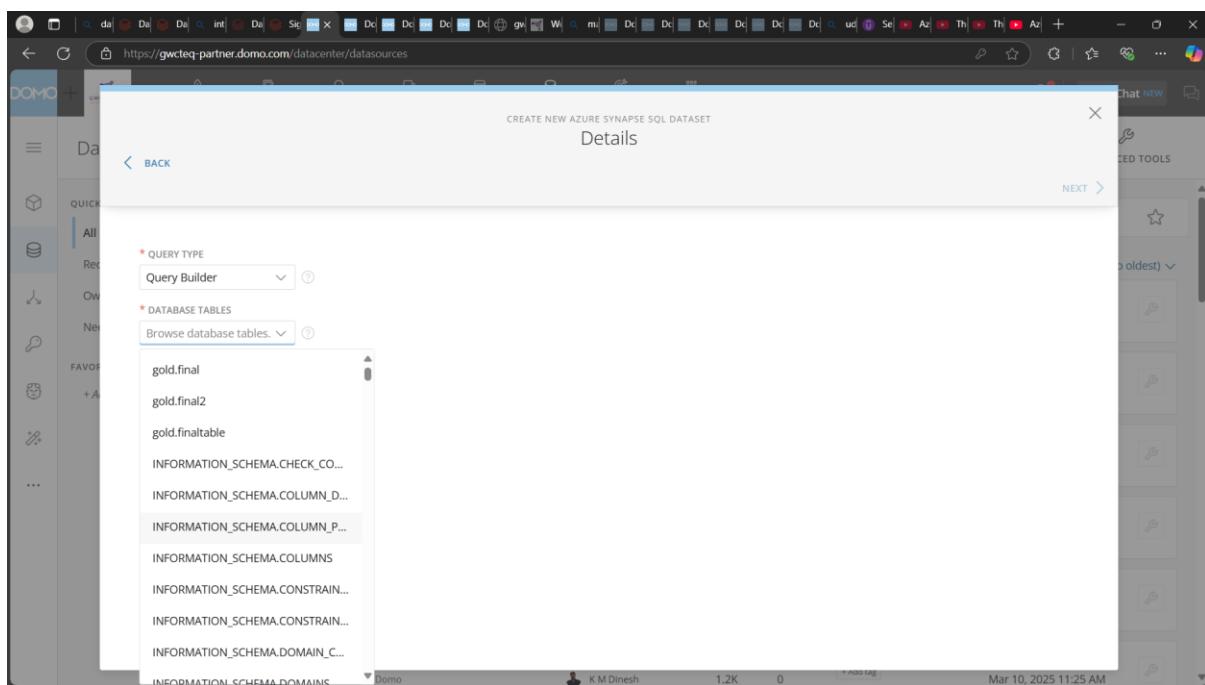
Details

* QUERY TYPE: Query Builder

* DATABASE TABLES: Browse database tables.

- gold.final
- gold.final2
- gold.finaltable
- INFORMATION_SCHEMA.CHECK_C...
- INFORMATION_SCHEMA.COLUMN_D...
- INFORMATION_SCHEMA.COLUMN_P...
- INFORMATION_SCHEMA.COLUMNS
- INFORMATION_SCHEMA.CONSTRAIN...
- INFORMATION_SCHEMA.CONSTRAIN...
- INFORMATION_SCHEMA.DOMAIN_C...
- INFORMATION_SCHEMA.DOMAINS

Domo K M Dinesh 1.2K 0 + Post tag Mar 10, 2025 11:25 AM



CREATE NEW AZURE SYNAPSE SQL DATASET

Details

BACK

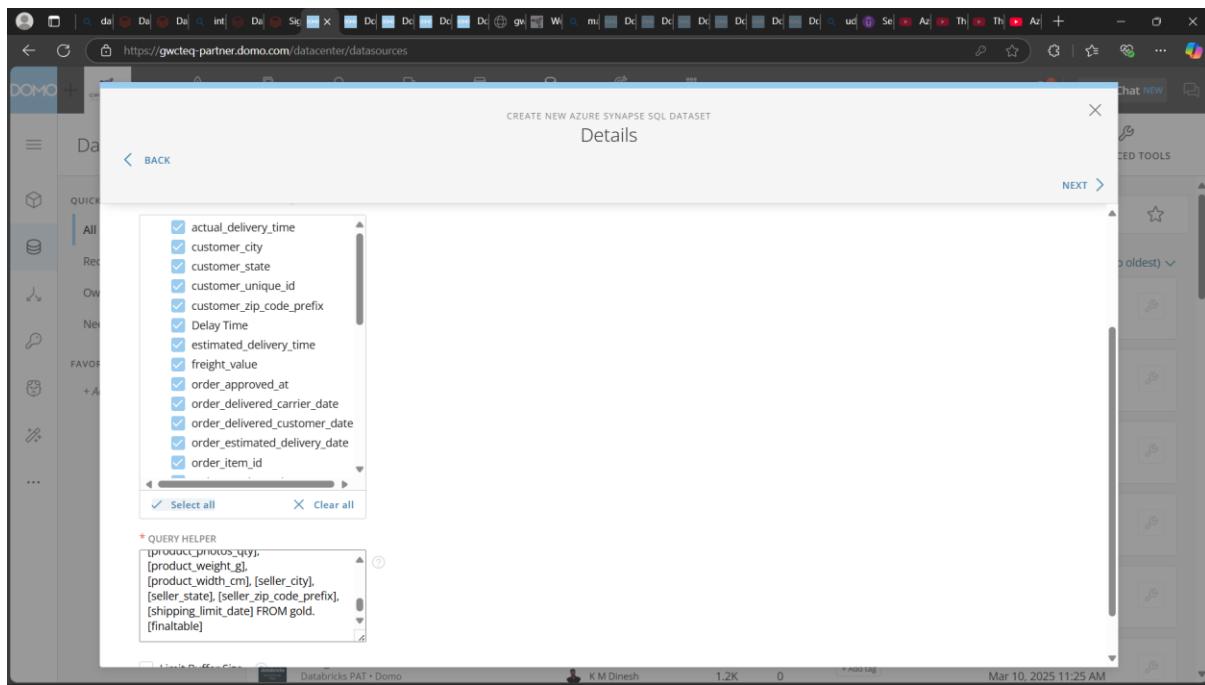
actual_delivery_time
customer_city
customer_state
customer_unique_id
customer_zip_code_prefix
Delay Time
estimated_delivery_time
freight_value
order_approved_at
order_delivered_carrier_date
order_delivered_customer_date
order_estimated_delivery_date
order_item_id

Select all Clear all

* QUERY HELPER

```
[product_name],  
[product_weight_g],  
[product_width_cm], [seller_city],  
[seller_state], [seller_zip_code_prefix],  
[shipping_limit_date] FROM gold  
[finaltable]
```

Databricks PAT • Domo KM Dinesh 1.2K 0 Mar 10, 2025 11:25 AM



https://gwcteq-partner.domo.com/datasources/12f80af6-316c-4eaf-a7f8-9da7ab37e4a8/details/overview

DOMO +

DATA

o list_synapse_dataset 0 rows

OVERVIEW AI READINESS DATA CARDS SETTINGS LINEAGE HISTORY PDP ALERTS AutoML OPEN WITH

olist_synapse_dataset

Azure Synapse SQL Connector - Domo - 0 columns - 0 rows - Importing... KM Dinesh

Direct Impact
Nothing in Domo uses this data

NO IMPACT

CREATE A VISUALIZATION

DOMO AI

CREATE APP

SHARE THIS DATASET

0 Cards powered

0 DataFlows created

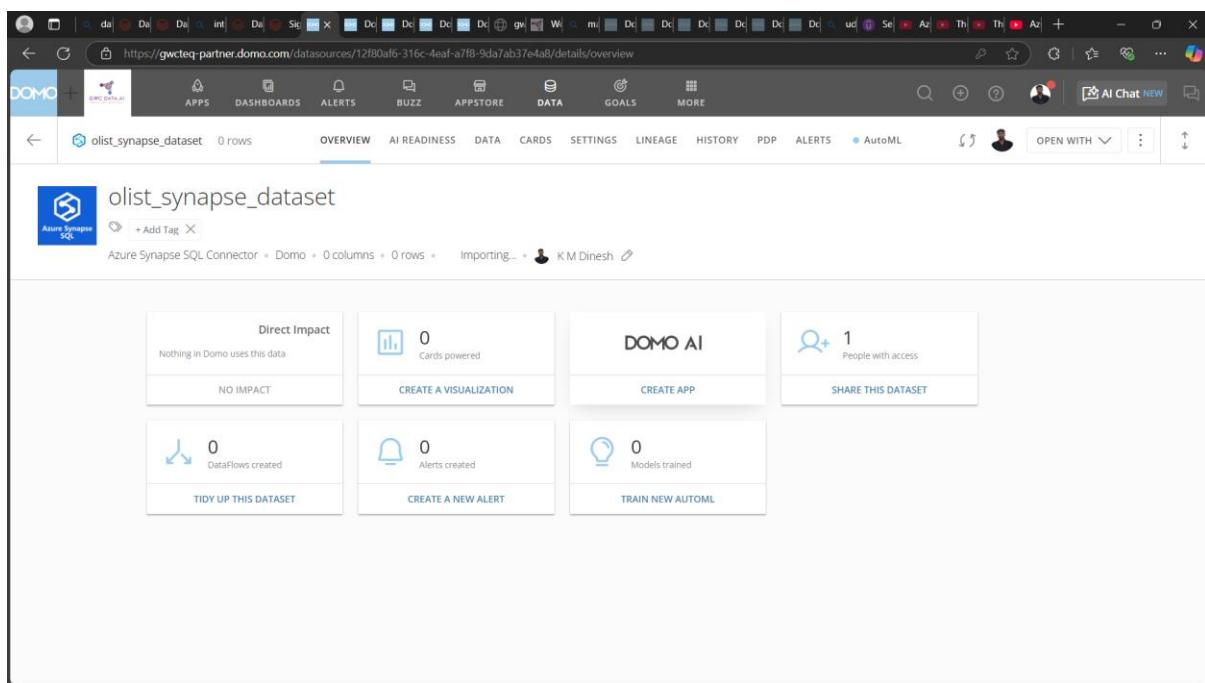
0 Alerts created

0 Models trained

TIDY UP THIS DATASET

CREATE A NEW ALERT

TRAIN NEW AUTOML



DOMO +  APPS DASHBOARDS ALERTS BUZZ APPSTORE DATA GOALS MORE

AI Chat NEW

olist_synapse_dataset 100 rows OVERVIEW AI READINESS DATA CARDS SETTINGS LINEAGE HISTORY PDP ALERTS AutoML OPEN WITH

Search Columns 100 rows

Delay Time actual_delivery_time customer_city customer_state customer_unique_id customer_zip_code_prefix

	123	123	123	AAC	AAC	AAC	Text
1	-6	10	campinas	SP	5fba1267f14b938597891c1cca2aec83		
2	-17	5	sao paulo	SP	813585153cd8b4b6c06e576b818ec985		
3	-32	7	sao paulo	SP	6bd4f06d5599d085d7a367cf49626155		
4	-20	22	belo horizonte	MG	aa28df52fd4d313847d126b4db11717		
5	-1	4	sao paulo	SP	09c7e80b16e4ae42787f124fa7b987f		
6	-14	8	curitiba	PR	60f7ebe1122afab7bfaf6c217ac8ea164		
7	-16	6	londrina	PR	4706af85da078301c35ace57729cbc6b		
8	-4	30	rio de janeiro	RJ	661b8ae47d627fb4e8570dcdb036710		
9	-10	8	osasco	SP	f51343c38ae261dab294c60577c21f5		
10	-5	7	sao carlos	SP	ab2731479851fc8019caa384772za2a0b		
11	-17	8	sao leopoldo	RS	b5f48e65c14e0e73846fb0f8611c64e		
12	-10	10	francisco morato	SP	a2694c5df183031534e4dd7411377		
13	-14	11	goianapolis	GO	5dbcfd1a3cc694bfec93ca15d884c9d		
14	-19	7	belo horizonte	MG	847ccb72903181d30dbf321c3d0b10ad		
15	-16	6	londrina	PR	4706af85da078301c35ace57729cbc6b		
16	-14	11	goianapolis	GO	5dbcfd1a3cc694bfec93ca15d884c9d		

Use this dataset to create dashboard

