

Data Engineering Project 2

Data Movement Pipeline

Name: Pradip Bochare

(Data Movement to Synapse using ADF)

1.1 Project Statement

Use Azure Data Factory to move data from an on-premises database to Azure Synapse Analytics & explain the process of the data movement pipeline

1.2 Project Overview

Welcome to the Data Movement Pipeline project on Azure! This project is designed to showcase how various Azure services can be utilized to ingest, store, transform data. This project provides a data engineering and analytical journey on the sample dataset. Starting with a CSV file which is stored in Azure Data Lake Gen 2 is copied into the Azure Synapse Analytics via Azure Data Factory. It's initially stored in Azure Data Lake Storage Gen2, then transformed in Azure synapse analytics. The final output is shown as data stored in ADLS gen 2 is copied to dedicated SQL pool in Azure Synapse Analytics.

1.3 Project Requirement:

The project aims to copy data from ADLS Gen 2 to Azure Synapse Analytics using Azure Data Factory. The data movement pipeline will enable users to visualize movement of data from ADLS to synapse.

A. Data Storage with ADLS Gen2 and Azure Synapse Analytics

- 1) Utilize Azure Synapse Analytics as the central data storage solution.
- 2) Implement data lakes and SQL pools to accommodate structured and unstructured data.
- 3) Design storage structures optimized for efficient querying and analysis of real-time data stream
- 4) Utilize Azure Data Lake Storage Gen2 (ADLS Gen2) to store large volumes of data efficiently and securely.
- 5) Design storage structures optimized for efficient querying, analysis, and storage of real-time data streams

B. Data Orchestration with Azure Data Factory

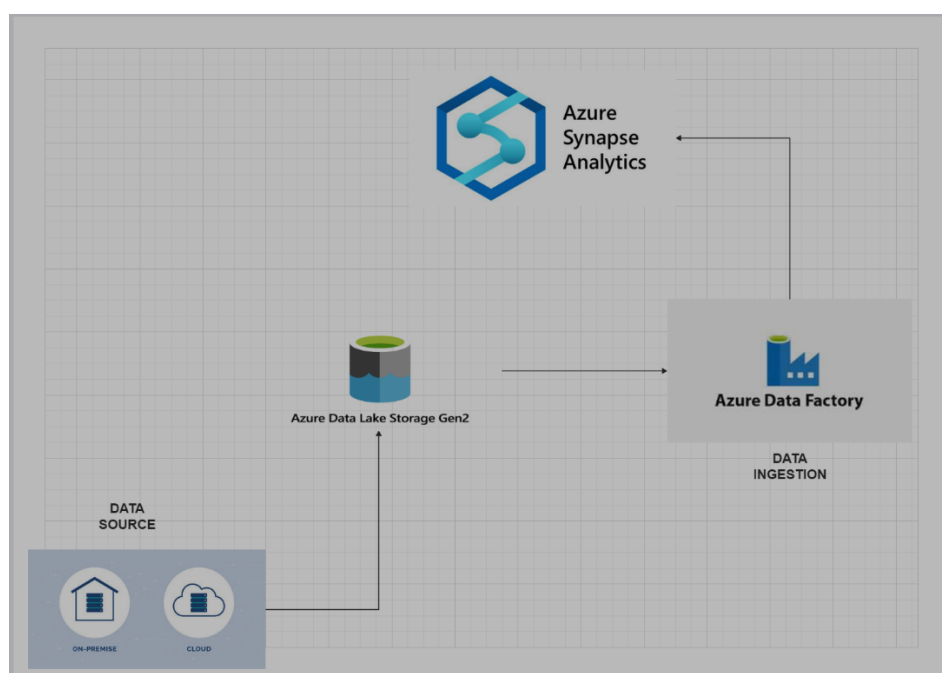
- 1) Implement Azure Data Factory to orchestrate data pipelines for ingesting, transforming, and loading data into Azure Synapse Analytics.
- 2) Utilize Data Factory pipelines to handle complex data transformations and data movement tasks efficiently.
- 3) Configure event-driven triggers to initiate data processing and analytics workflows in real-time.

C. Integration and data connectivity

- 1) Integrate the Azure Data Factory and Azure Synapse Analytics to fetch live data streams and analytics results.
- 2) Establish secure connections between dedicated SQL pools and backend data sources to ensure data integrity and confidentiality.
- 3) Implement efficient data refresh mechanisms to keep the data up-to-date with the latest changes in storage account

2.1 Architectural Diagram

The architecture for the project involving building data movement pipeline using Azure Data Factory for data movement from ADLS Gen 2 to Azure Synapse Analytics.



3.1 Tools Used in Data Movement Pipeline

To build a data movement pipeline using ADLS gen 2 for data storage, Azure Synapse Analytics, Azure Data Factory for orchestrating data pipelines, the following tools and services are utilized:

1)Azure Data Lake Storage Gen2(ADLS Gen2)

ADLS Gen2 provides scalable, secure, and cost-effective storage for big data analytics workloads. It allows storage of structured, semi-structured, and unstructured data in a distributed file system. ADLS Gen2 integrates seamlessly with other Azure services, making it suitable for real-time analytics scenarios.

2)Azure Synapse Analytics

Formerly known as Azure SQL Data Warehouse, Azure Synapse Analytics is a cloud-based analytics service that brings together enterprise data warehousing and big data analytics. It provides capabilities for data storage, data integration, data warehousing, and big data analytics in a single platform. Azure Synapse Analytics enables the storage and analysis of large volumes of data in real-time, making it ideal for building real-time analytics dashboards.

4)Azure Data Factory

Azure Data Factory is a cloud-based data integration service that allows users to create, schedule, and orchestrate data pipelines. It supports the movement and transformation of data between various sources and destinations, including ADLS Gen2, Azure Synapse Analytics, and Azure Databricks. Azure Data Factory enables the creation of real-time data pipelines for ingesting, processing, and transforming data for analytics purposes.

5)Dataset Used

This contains the details of CSV file which is downloaded from open-source platform which is athletes.csv. This dataset contains the details of the Athletes.

Prerequisite

1. **Azure Data Factory:** For data ingestion from GitHub.
2. **Azure Data Lake Storage Gen2:** As the primary data storage solution.
3. **Azure Synapse Analytics:** To perform in-depth data analytics.

3.2 Execution Overview

The execution process for data movement pipeline building using ADLS Gen2, Azure Synapse Analytics, Azure Data Factory:

1)Azure Environment Setup

Provision Azure services including ADLS Gen2, Azure Synapse Analytics, Azure Databricks, Azure Data Factory, and Power BI. Configure networking, security, and access controls based on project needs.

2)Data Ingestion with Data Factory

Create data ingestion pipelines in Azure Data Factory to ingest data from various sources into ADLS Gen2 and Azure Synapse Analytics. Implement real-time data ingestion where applicable.

3)Data store and management

Optimize data storage structures and configurations in ADLS Gen2 and Azure Synapse Analytics. Implement partitioning, indexing, and compression techniques for efficient storage and querying.

4.1 Implementation and Development Process

To get started with this project, follow these steps:

- 1) **Infrastructure setup and configuration:** Provision Azure services including ADLS Gen2, Azure Synapse Analytics, Azure Data Factory. Configure networking, security, and access controls based on project requirements.
- 2) **Data Modelling and Schema Design:** Design data models and schemas for storing and processing data in ADLS Gen2 and Azure Synapse analytics. Define data structures and formats for efficient real-time analytics processing
- 3) **Data Ingestion:** Use Azure Data Factory to configure data ingestion from your chosen data sources. Define the data movement and transformation activities required to bring the data into the Azure ecosystem. Implement data ingestion pipelines using Azure Data Factory to ingest data from multiple sources including IoT devices, applications, and databases into ADLS Gen2 and Azure Synapse. Ensure real-time or near real-time data ingestion capabilities to enable timely analytics
- 4) **Data Storage and management optimization:** Optimize data storage structures and configurations in ADLS Gen2 and Azure Synapse Analytics for efficient querying, storage, and retrieval. Implement partitioning, indexing, and compression techniques to improve performance and scalability.
- 5) **Integration and connectivity:** Establish seamless integration between Azure Synapse Analytics, ADLS Gen2, Azure Data Factory. Configure data connectors, APIs, and authentication mechanisms for secure data exchange and connectivity.
- 6) **Testing and Quality Assurance:** Conduct comprehensive testing of data pipelines, analytics algorithms, and dashboard functionality. Perform unit tests, integration tests, and end-to-end tests to validate data accuracy, system performance, and user experience.
- 7) **Development and production rollout:** Deploy the real-time analytics dashboard to the production environment. Monitor system performance, data integrity, and user interactions during the initial rollout phase. Address any issues or bugs identified during deployment and ensure smooth operation of the dashboard.

- 8) **Documentation and User Training:** Document the architecture, design decisions, configuration settings, and implementation details of the real-time analytics dashboard. Provide user documentation and training materials to educate users on how to interact with the dashboard and interpret analytics insights effectively.

By following this implementation and development process, you can build a robust data movement pipeline that leverages the capabilities of ADLS Gen2, Azure Synapse Analytics, Azure Data Factory to enable data-driven decision-making and insights generation in real-time

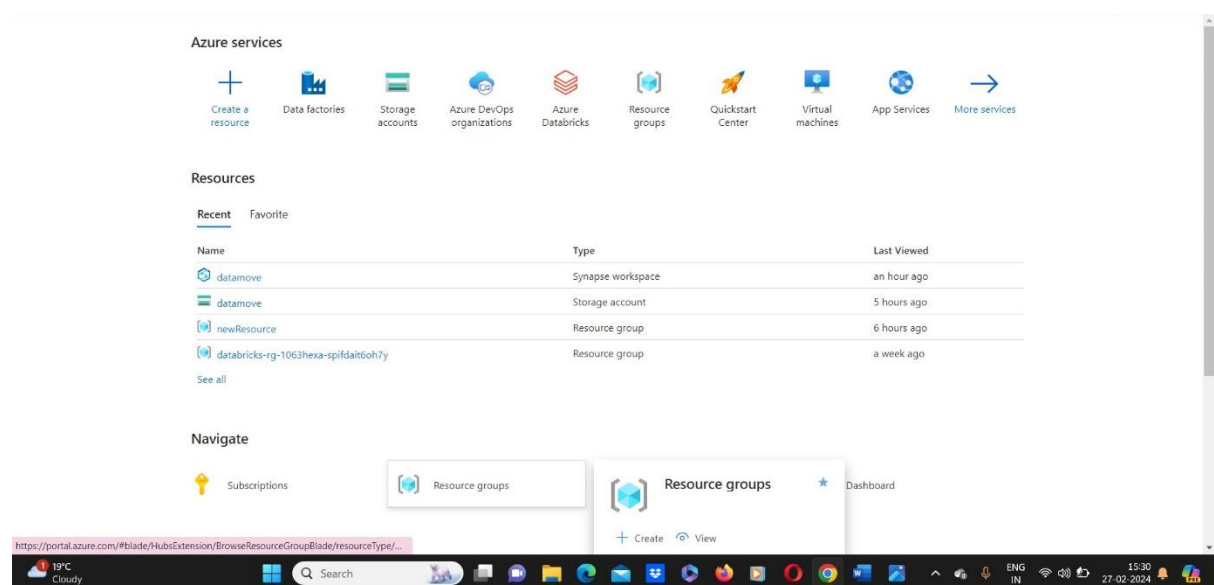
5.1 Tasks Performed on Data Movement Pipeline Project

I used the local dataset and tried to use every important feature Azure had to offer. I **integrated** the data from my computer to a **data lake** by using **Microsoft Azure Data Factory**. A pipeline was built to integrate all data and **validated** to check for any errors.

Workflow

Initial Setup

1. Create Azure Free Subscription account



2. Create a Resource Group to house and manage all the Azure resources associated with this project.
3. Within the created resource group, set up a storage account. This is specifically configured to leverage Azure Data Lake Storage (ADLS) Gen2 capabilities.

The screenshot shows the 'Create a storage account' wizard in the Microsoft Azure portal. The 'Basics' tab is selected, showing the following configuration:

- Subscription:** Azure subscription 1
- Resource group:** rg-azuser1077_mml.local-IWFLY (with a 'Create new' link)
- Storage account name:** hexadeb1077
- Region:** (Asia Pacific) Central India (with a 'Deploy to an edge zone' link)
- Performance:** Standard: Recommended for most scenarios (general-purpose v2 account) (selected)
- Redundancy:** Geo-redundant storage (GRS) (selected)
- Make read access to data available in the event of regional unavailability:** (checked)

At the bottom, there are 'Previous', 'Next', and 'Review + create' buttons. The 'Review + create' button is highlighted in blue.

The screenshot shows the 'Overview' page for the storage account 'hexadeb1077'. The page displays various properties and settings:

- Resource group:** rg-azuser1077_mml.local-IWFLY
- Location:** centralindia
- Primary/Secondary Location:** Primary: Central India, Secondary: South India
- Subscription:** Azure subscription 1
- Subscription ID:** 984f097c-963c-4eb6-a20d-839457ae9f08
- Disk state:** Primary: Available, Secondary: Available
- Performance:** Standard
- Replication:** Read-access geo-redundant storage (RA-GRS)
- Account kind:** StorageV2 (general purpose v2)
- Provisioning state:** Succeeded
- Created:** 26/02/2024, 23:04:41

The 'Properties' tab is selected, showing a table of capabilities and their status:

Capability	Status
Hierarchical namespace	Enabled
Default access tier	Hot
Blob anonymous access	Disabled
Blob soft delete	Enabled (7 days)
Container soft delete	Enabled (7 days)
Versioning	Disabled
Change feed	Disabled
NFS v3	Disabled
SFTP	Disabled
Storage tasks assignments	None

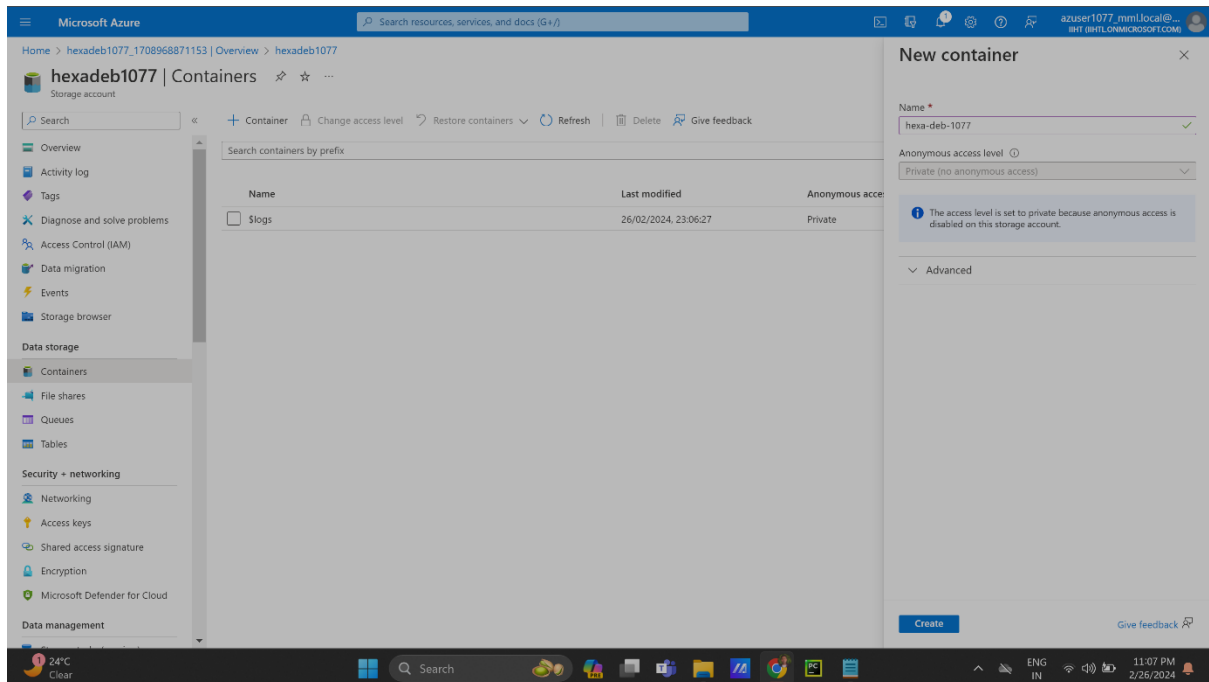
The 'Security' section shows the following settings:

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

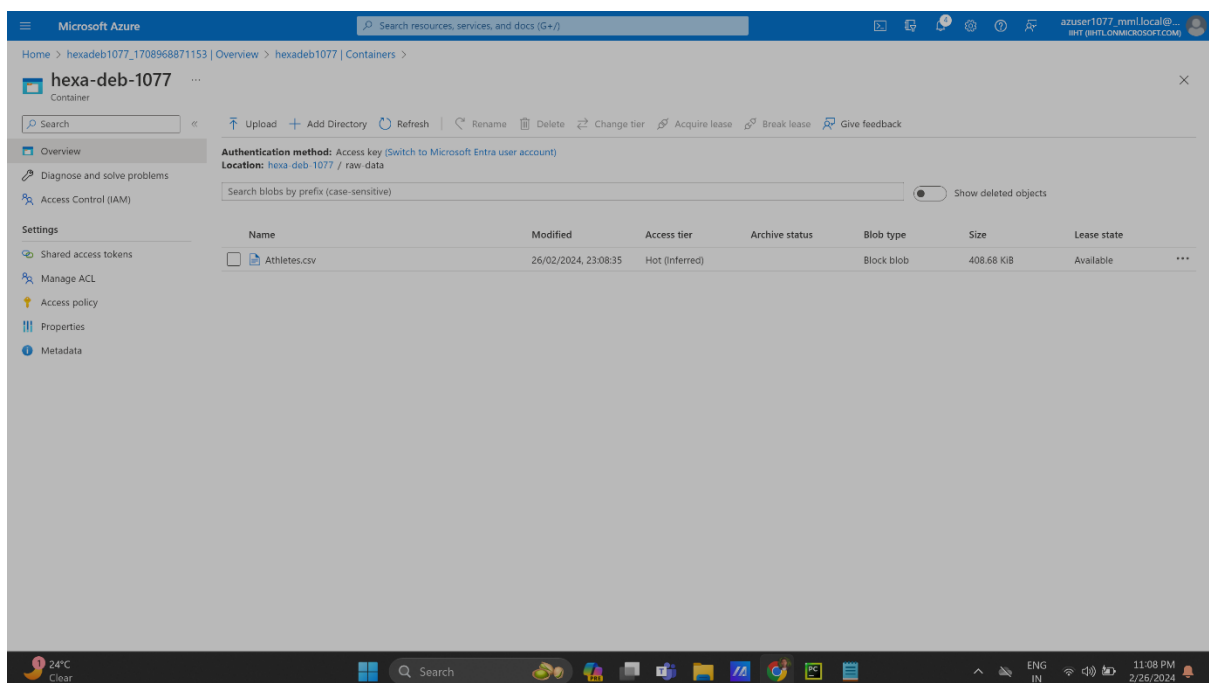
The 'Networking' section shows the following settings:

- Allow access from: All networks
- Number of private endpoint connections: 0
- Network routing: Microsoft network routing
- Access for trusted Microsoft services: Yes
- Endpoint type: Standard

4. Create a Container inside this storage account to hold the project's data. One directory 'raw-data' is created to store raw data.



5. Uploaded one CSV file into the container



Setting Up and Using Azure Synapse Analytics

1. Creating a Synapse Analytics Workspace.
2. Within Workspace navigate to the "Data" section, choose "Lake Database" and create a Database "hexadeb1077as"
3. Creating Table from Data Lake from the Transformed Data folder within your ADLS Gen2 storage.

Creating Azure Synapse Analytics Workspace

The screenshot shows the 'Create Synapse workspace' wizard in the Azure portal. The interface is in English and the user is logged in as 'azuser1077_mml.local@...'. The wizard has tabs for 'Basics', 'Security', 'Networking', 'Tags', and 'Review + create'. The 'Basics' tab is active, showing the 'Project details' section. Below this, there are fields for 'Subscription' (Azure subscription 1), 'Resource group' (rg-azuser1077_mml.local-WFLY), and 'Managed resource group' (Enter managed resource group name). The 'Workspace details' section follows, with fields for 'Workspace name' (hexadeb1077as), 'Region' (Central India), and 'Select Data Lake Storage Gen2' (From subscription). Below these are fields for 'Account name' (hexadeb1077) and 'File system name' (hexa-deb-1077). At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next: Security >'. The Windows taskbar at the bottom shows the date as 2/25/2024 and the time as 11:11 PM.

Microsoft Azure

Search resources, services, and docs (G+/J)

Home > Azure Synapse Analytics >

Create Synapse workspace

*** Basics** * Security Networking Tags Review + create

Create a Synapse workspace to develop an enterprise analytics solution in just a few clicks.

Project details

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

Subscription * Azure subscription 1

Resource group * rg-azuser1077_mml.local-WFLY
[Create new](#)

Managed resource group Enter managed resource group name

Workspace details

Name your workspace, select a location, and choose a primary Data Lake Storage Gen2 file system to serve as the default location for logs and job output.

Workspace name * hexadeb1077as

Region * Central India

Select Data Lake Storage Gen2 * ☒ From subscription ☐ Manually via URL

Account name * hexadeb1077
[Create new](#)

File system name * hexa-deb-1077
[Create new](#)

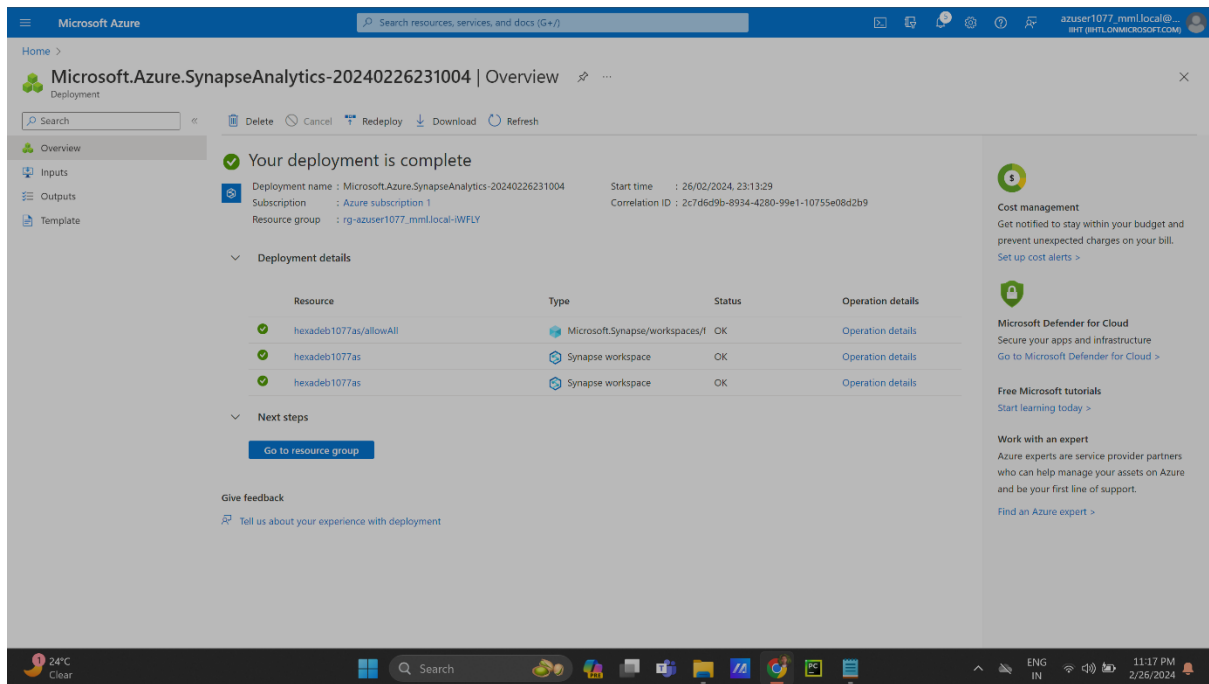
[Review + create](#) < Previous Next: Security >

24°C Clear

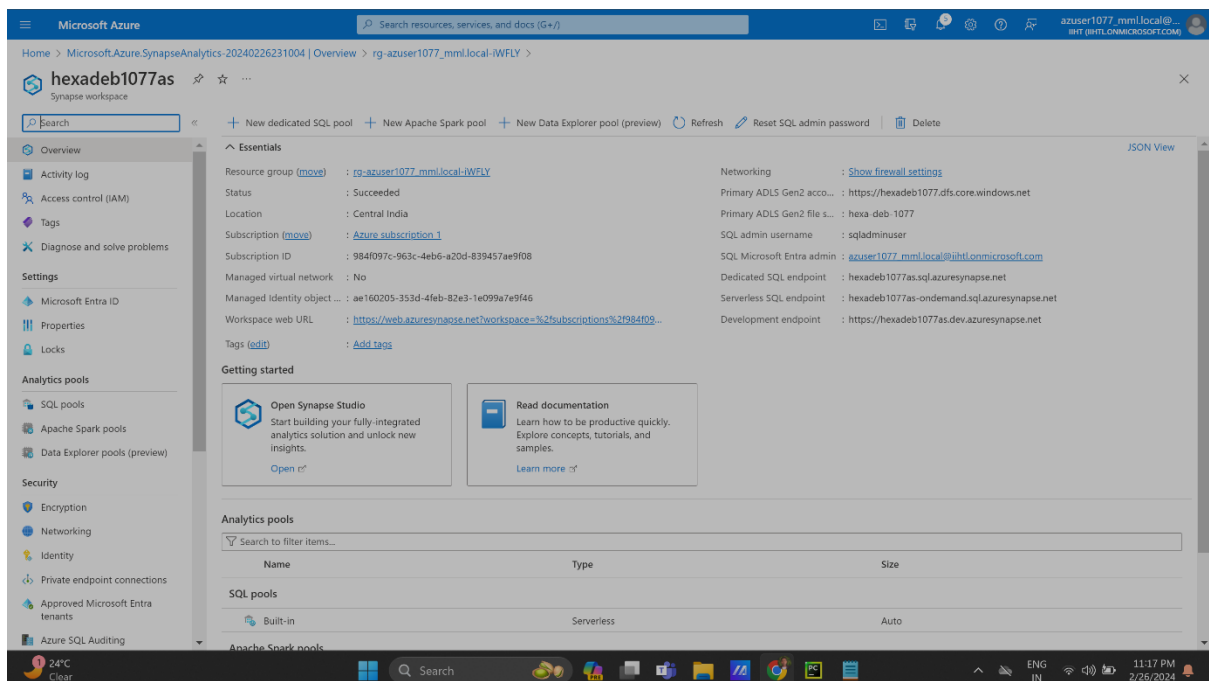
Search

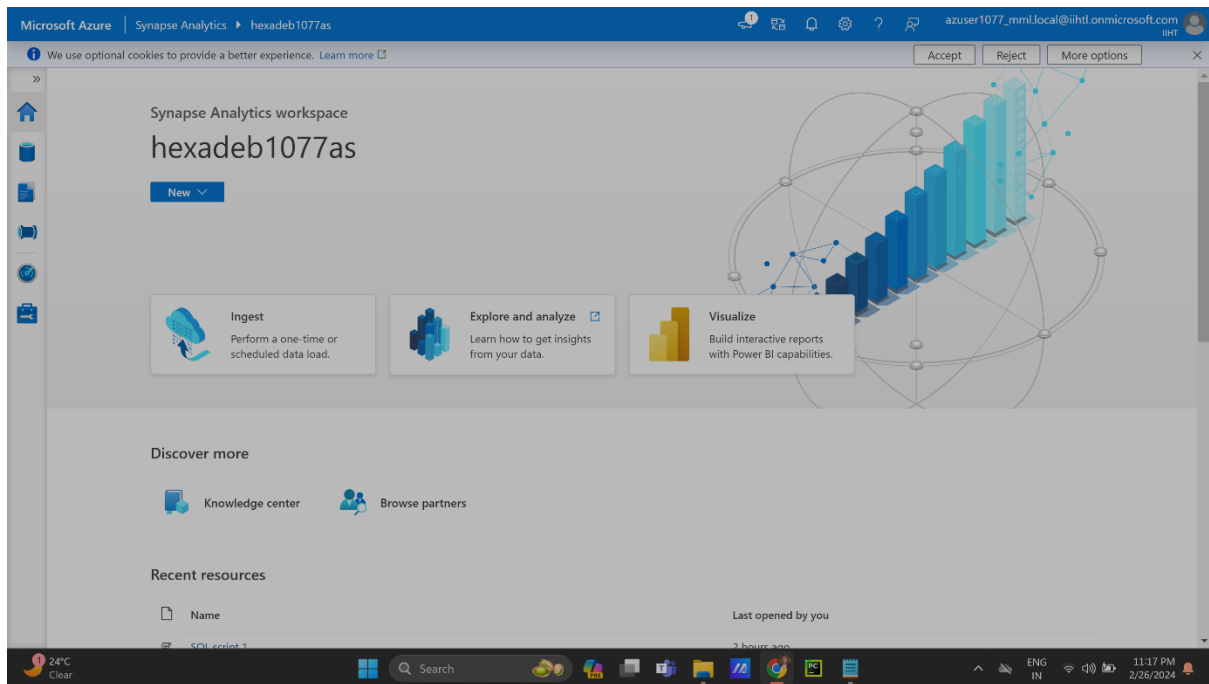
ENG IN

11:11 PM 2/25/2024

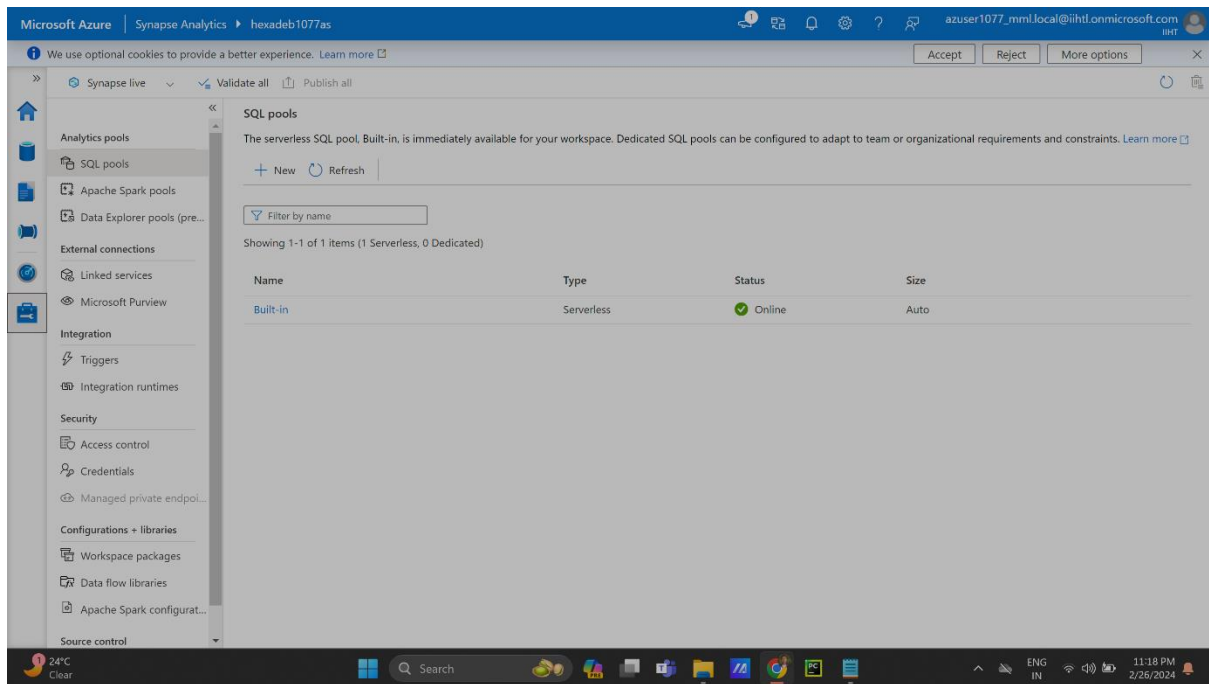


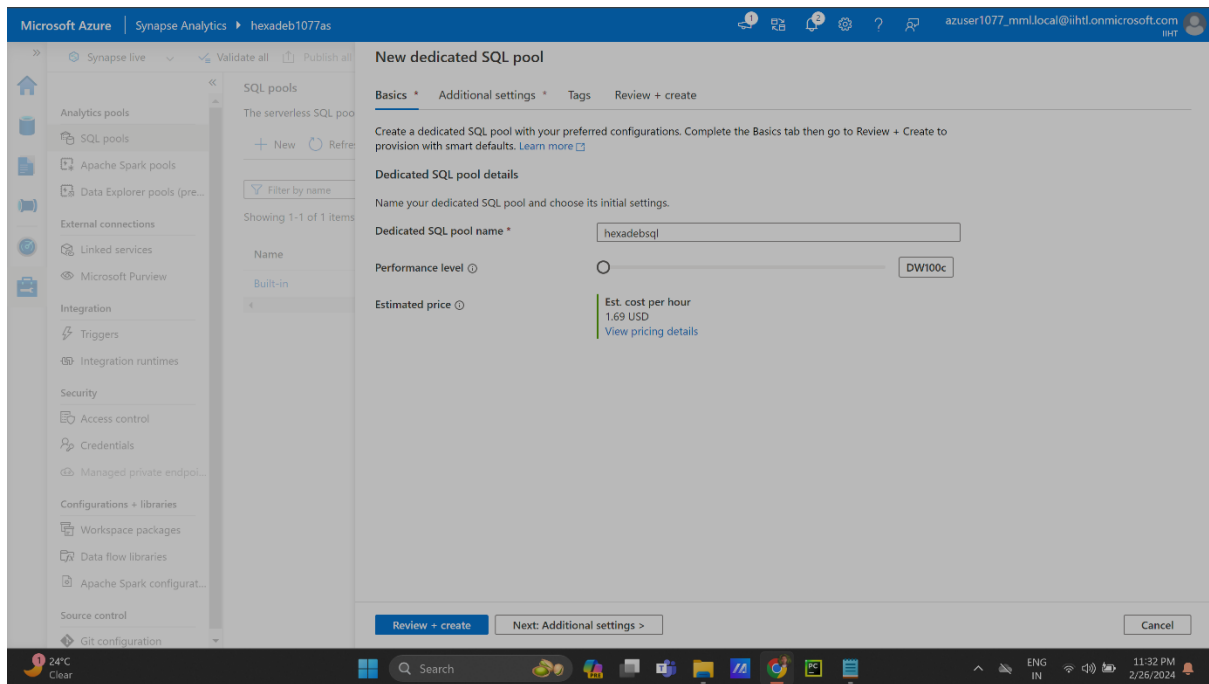
Overview of Synapse Workspace



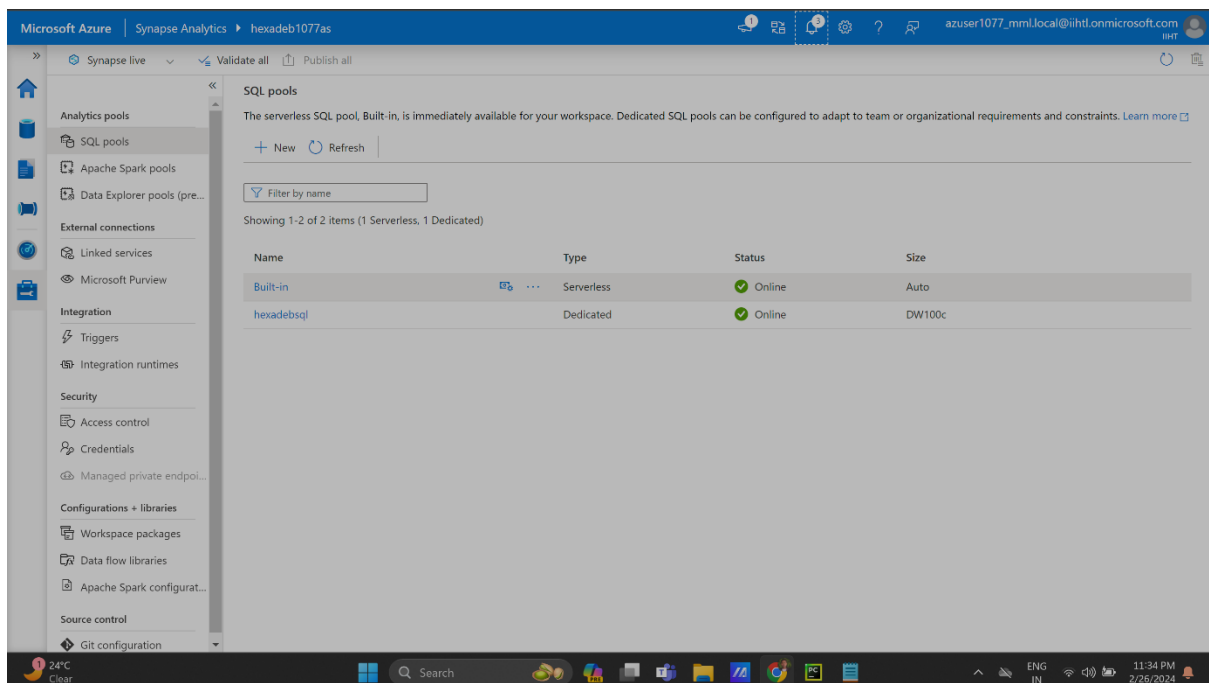


Creating Dedicated SQL pool to move data from ADLS Gen 2 to Synapse





Dedicated SQL pool is created



Data Ingestion using Azure Data Factory

1. Begin by creating an Azure Data Factory workspace within the previously established resource group.
2. After setting up the workspace, launch the Azure Data Factory Studio.
3. Within the studio, initialize a new data integration pipeline. Now use the task Copy Data to move data efficiently between various supported sources and destinations.
4. Configuring the Data Source with ADLS Gen 2 as we are using ADLS template to get data.
5. Establishing the Linked Service for source.
6. Configuring the File Format for and setting up the Linked Service Sink.
7. Repeat above steps to load the dataset.
8. You can connect all the copy data activity together and run them all at once.

Creating Azure Data Factory

The screenshot shows the 'Create Data Factory' wizard in the Microsoft Azure portal. The 'Project details' section is active, showing fields for Subscription, Resource group, Name, Region, and Version. The 'Instance details' section is also visible.

Project details

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

Subscription *

Resource group *
[Create new](#)

Instance details

Name *

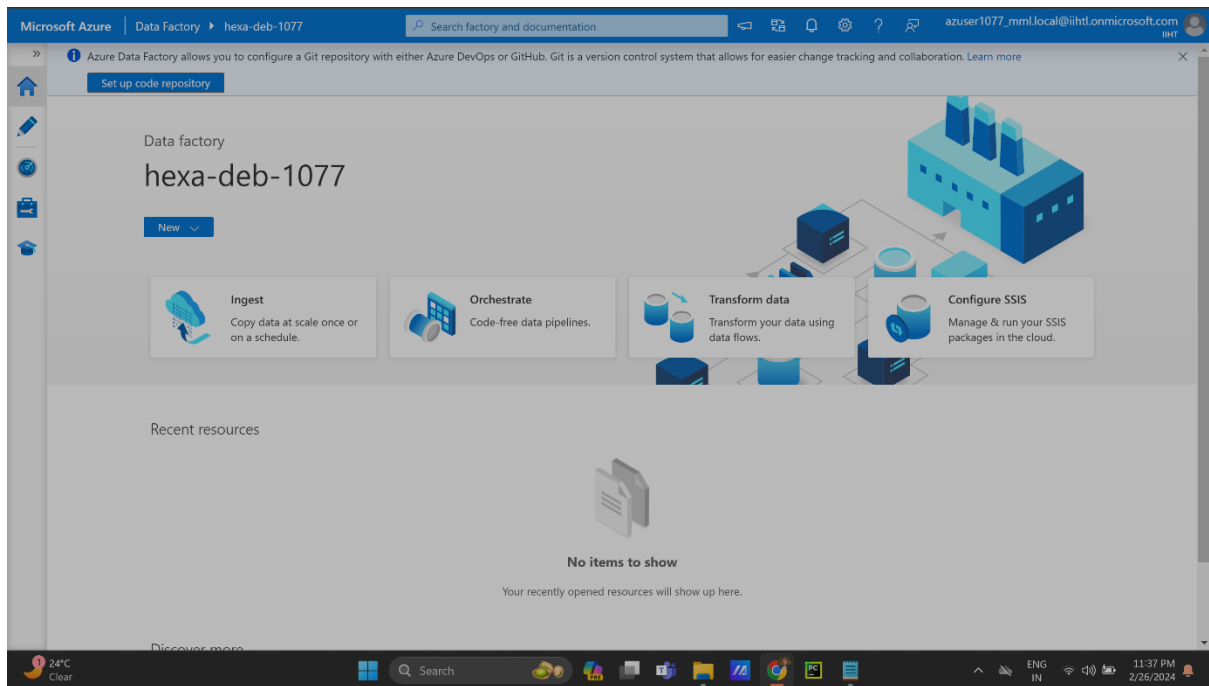
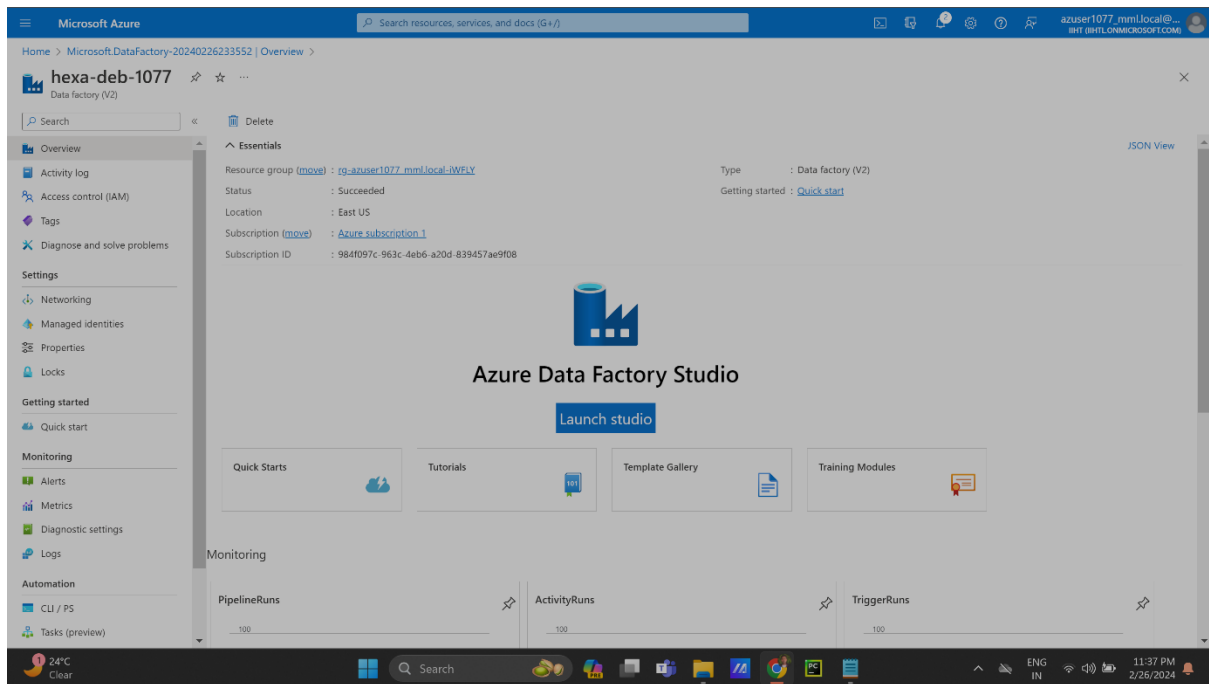
Region *

Version *

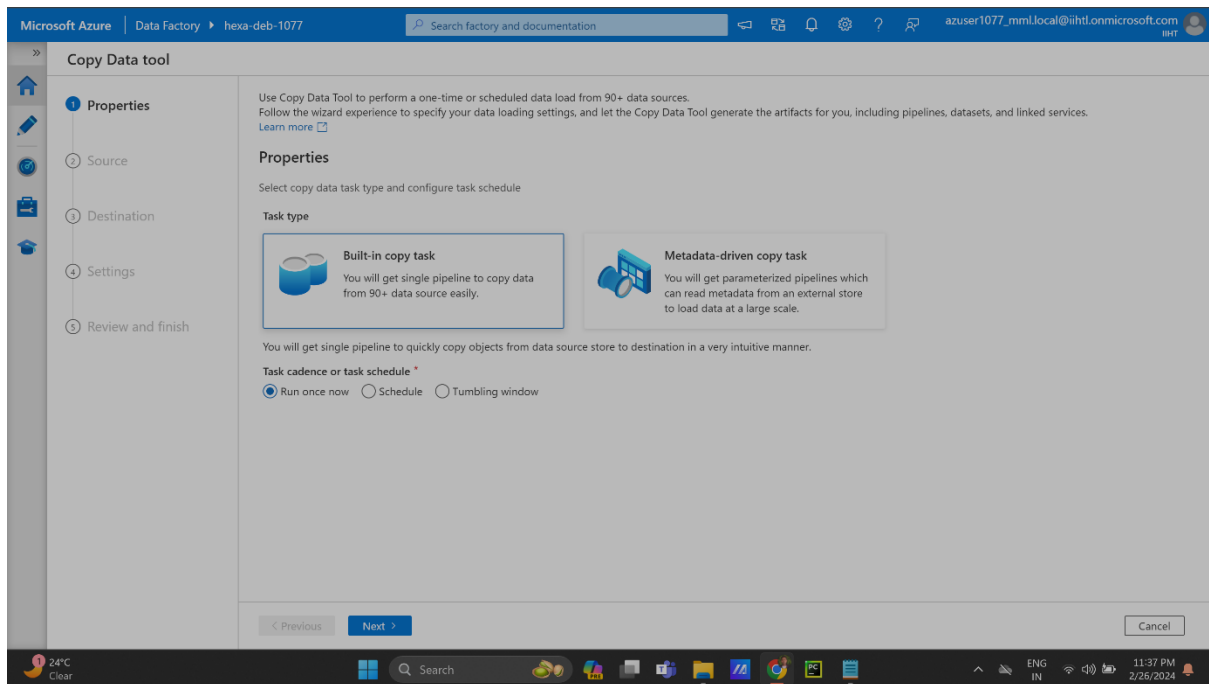
Navigation: [Previous](#) [Next](#) [Review + create](#)

Footer: [Give feedback](#)

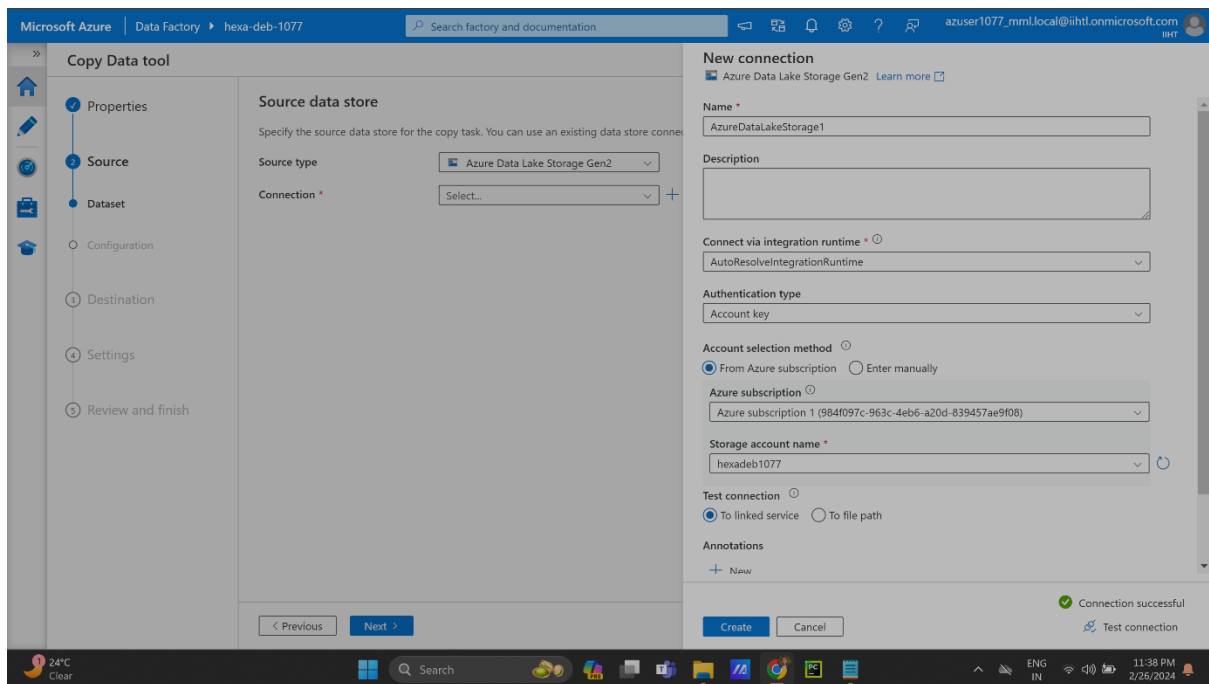
Overview of Azure Data Factory



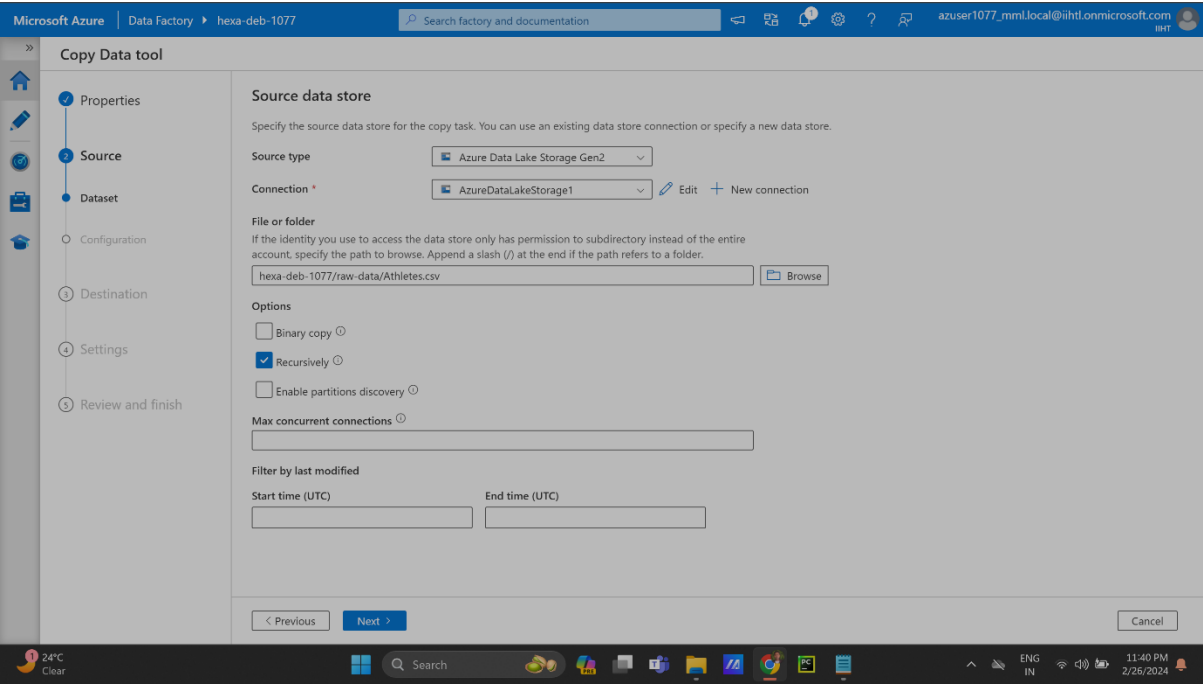
Starting to Ingest Data in Azure Data Factory to make pipeline



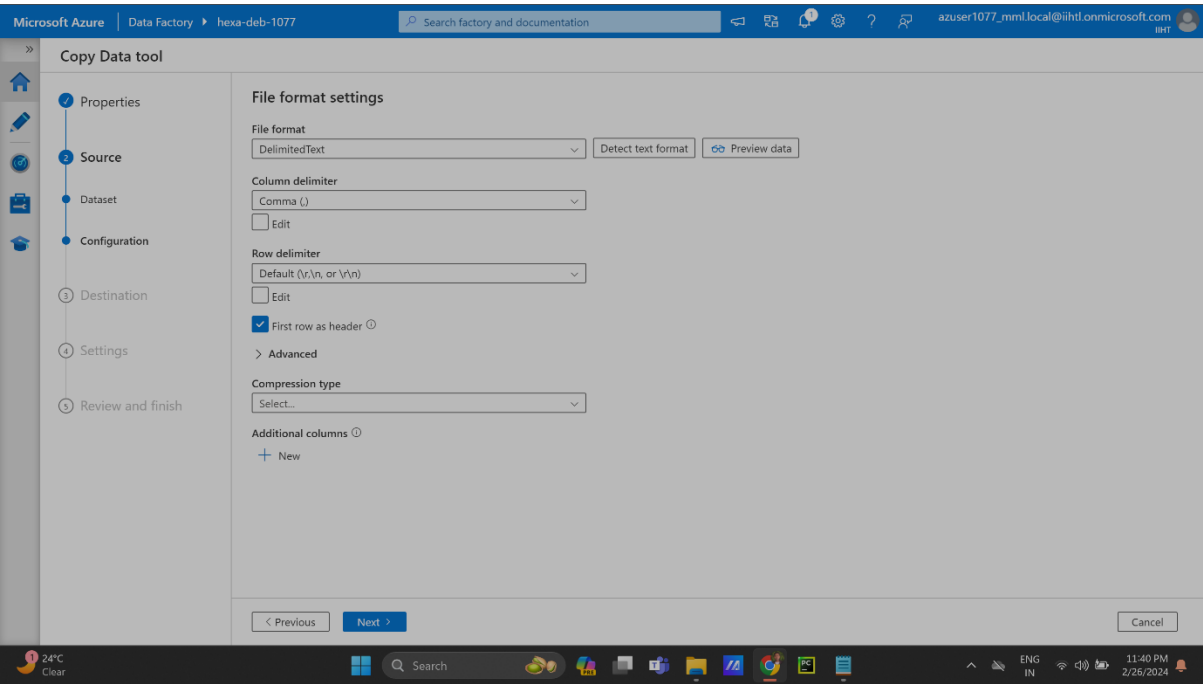
Setting up Source for Data



Source Data Store



Configuration of Source Data



Preview of Source Data

The screenshot shows the 'Copy Data tool' interface in Microsoft Azure Data Factory. The 'Preview data' window is open, displaying a table of data from the source. The table has three columns: PersonName, Country, and Discipline. The data is as follows:

PersonName	Country	Discipline
AALERUD Katrine	Norway	Cycling Road
ABAD Nestor	Spain	Artistic Gymnastics
ABAGNALE Giovanni	Italy	Rowing
ABALDE Alberto	Spain	Basketball
ABALDE Tamara	Spain	Basketball
ABALO Luc	France	Handball
ABAROA Cesar	Chile	Rowing
ABASS Abobakr	Sudan	Swimming
ABBASALI Hamideh	Islamic Republic of Iran	Karate

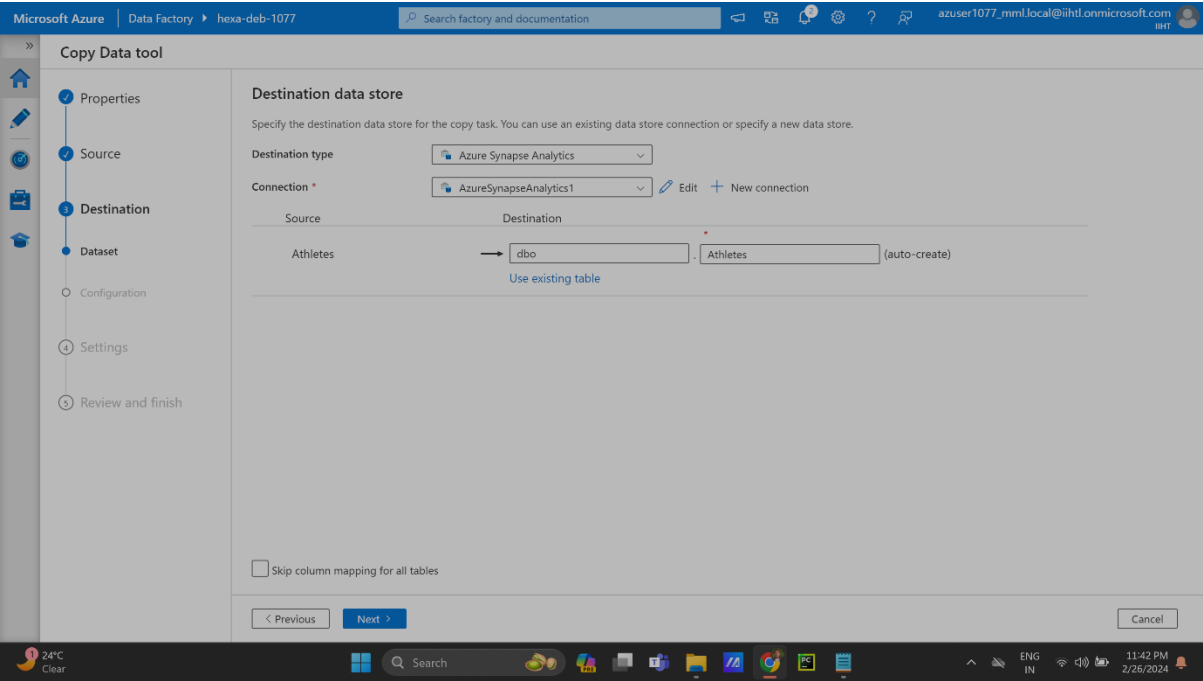
Setting up Destination

The screenshot shows the 'New connection' window in the 'Copy Data tool' interface. The window is for setting up a new connection to Azure Synapse Analytics. The fields are as follows:

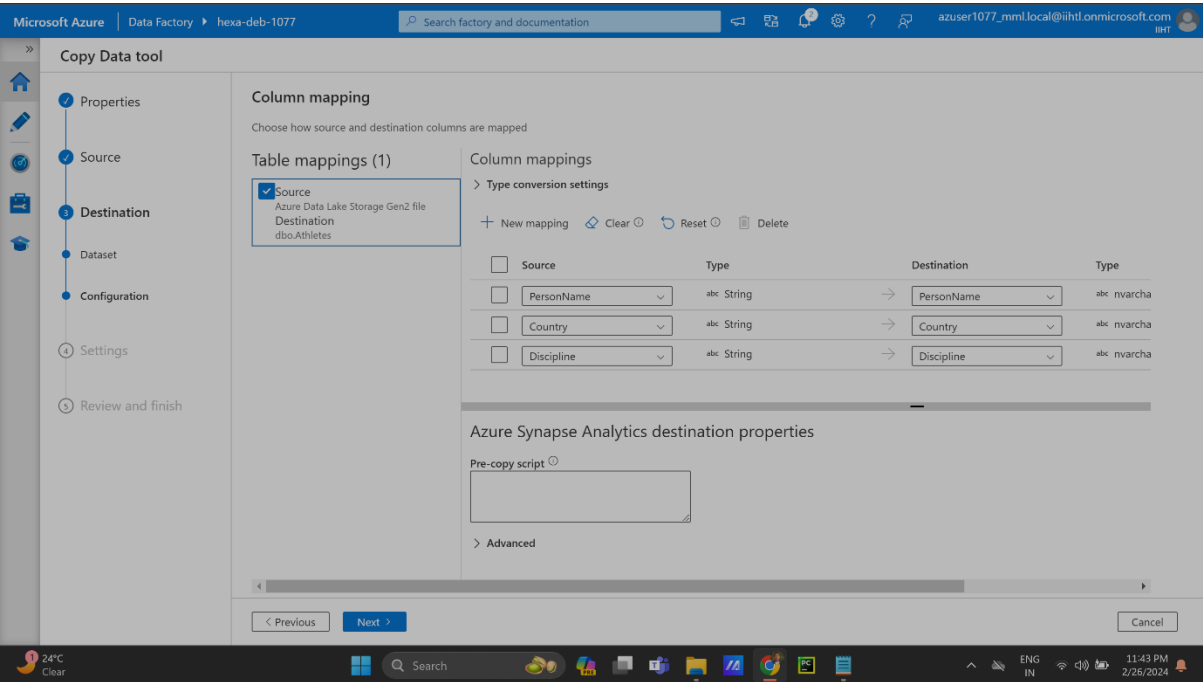
- Name: AzureSynapseAnalytics1
- Description: (empty)
- Connect via integration runtime: AutoResolveIntegrationRuntime
- Account selection method: From Azure subscription
- Azure subscription: Azure subscription 1 (984f097c-963c-4eb6-a20d-839457ae9f08)
- Server name: hexadeb1077as (Synapse workspace)
- Database name: hexadeb1077as (Synapse workspace)
- SQL pool: hexadeb1077as (Synapse workspace)

The 'Connection successful' message is displayed at the bottom right of the window.

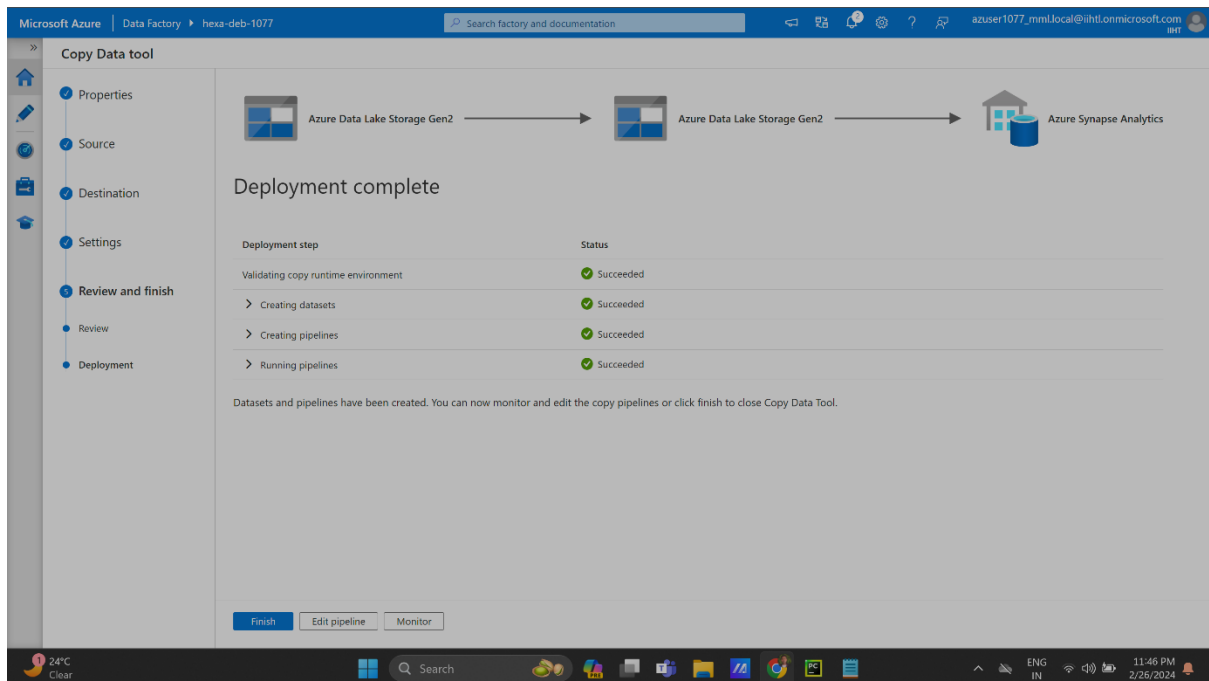
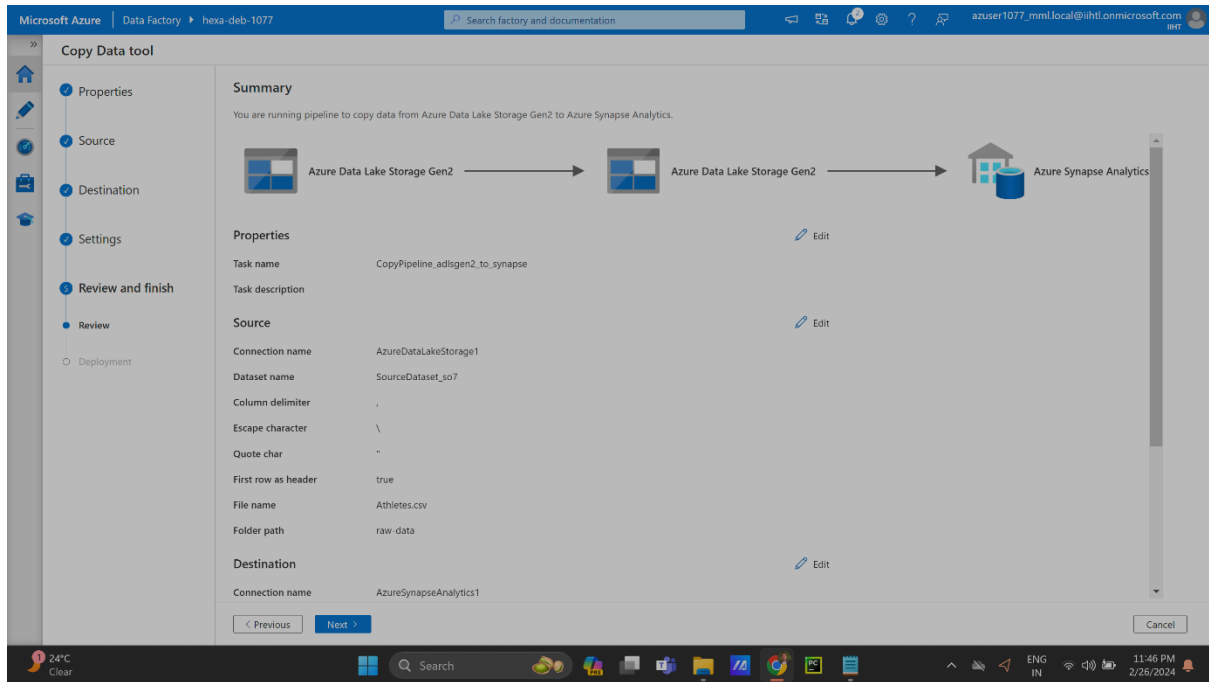
Destination Data Store



Configuration For Destination



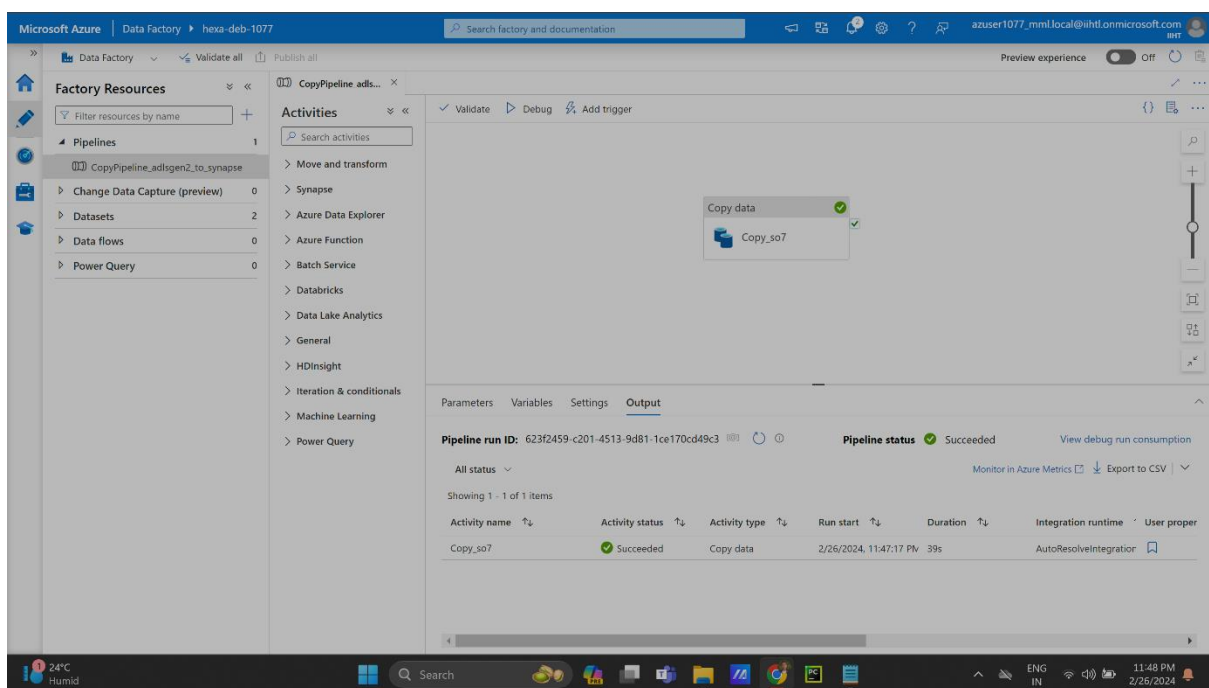
Deployment

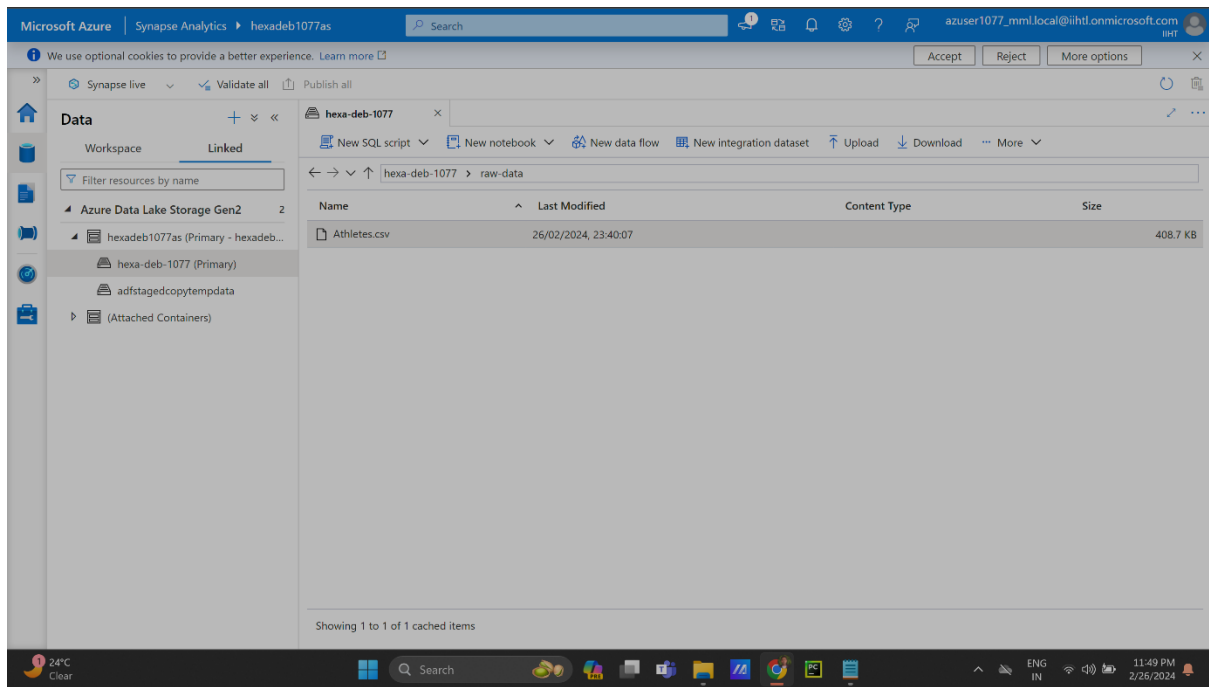


6.1 FINAL OUTPUT SCREENSHOT

The final output screenshot of the data movement pipeline:

The final output is data movement pipeline created using Azure Data Factory which helps to move data from ADLS Gen 2 to dedicated SQL pool in Azure Synapse Analytics. The output shows data pipeline running successful in Azure Data Factory. The final output shows the data present in ADLS Gen 2 copied in Azure Synapse Analytics dedicated SQL pool.





CONCLUSIONS

The conclusion of the project to build a data movement pipeline is the project successfully achieved its objective of creating data movement pipeline that moves data from ADLS Gen 2 to Azure Synapse Analytics using Azure Data Factory. Azure Data Factory effectively orchestrated data pipelines, ensuring seamless and reliable data ingestion from various sources into ADLS Gen2 and Azure Synapse Analytics. The Data Movement Pipeline provides the users to move data from ADLS Gen 2 to Azure Synapse using Data factory. This pipeline can run recursively to copy data which is added in storage account frequently.

