

Name : Parth Nandedkar

Date : 21 Feb 2024

Topics : Azure Databricks

Batch : Data Engineering Batch-1

Q3.Execute & explain, Azure datafactory and its copy activity.

Azure Data Factory is a cloud-based data integration service that allows you to create, schedule, and orchestrate data workflows at scale. It enables you to **collect, transform, and move data** from various sources to different destinations, both within Azure and across on-premises and other cloud environments. Here's an overview of Azure Data Factory and its key components:

Pipelines:

Pipelines are the core construct in Azure Data Factory. They represent a series of data processing steps that perform operations on data. These steps can include data movement, transformation, data loading, and data orchestration activities.

Pipelines are composed of activities, which are the building blocks for performing tasks like copying data from a source to a destination, transforming data using Azure services like HDInsight or Azure Databricks, executing SQL scripts, etc.

Activities:

Activities are the individual processing steps within a pipeline. They represent the actions that are performed on data, such as copying data from a source to a destination, running a Hive query, executing a stored procedure, etc.

Azure Data Factory provides a wide range of built-in activities for common data integration tasks, and you can also create custom activities using Azure Functions or Azure Batch if needed.

Datasets:

Datasets represent the structure of the data being processed within Azure Data Factory. They define the schema and location of the data, whether it's in files, databases, or other data stores.

Azure Data Factory supports various types of datasets, including Azure Blob Storage, Azure Data Lake Storage, Azure SQL Database, Azure Synapse Analytics, and many others.

Linked Services:

Linked Services are connections to external data sources or destinations used by Azure Data Factory activities. They contain the information needed to connect to the data stores, such as connection strings, credentials, and authentication methods.

Linked Services are defined separately from datasets and can be reused across pipelines.

Triggers:

Triggers are used to schedule the execution of pipelines in Azure Data Factory. There are different types of triggers available, including schedule-based triggers, event-based triggers, and tumbling window triggers. Schedule-based triggers allow you to specify a recurrence pattern for running pipelines at regular intervals, while event-based triggers respond to events such as the arrival of new data or the completion of a data processing task.

Integration Runtimes:

Integration Runtimes are compute environments used by Azure Data Factory to execute data integration tasks. They provide the resources and capabilities needed to connect to data sources, perform data transformations, and move data between different environments.

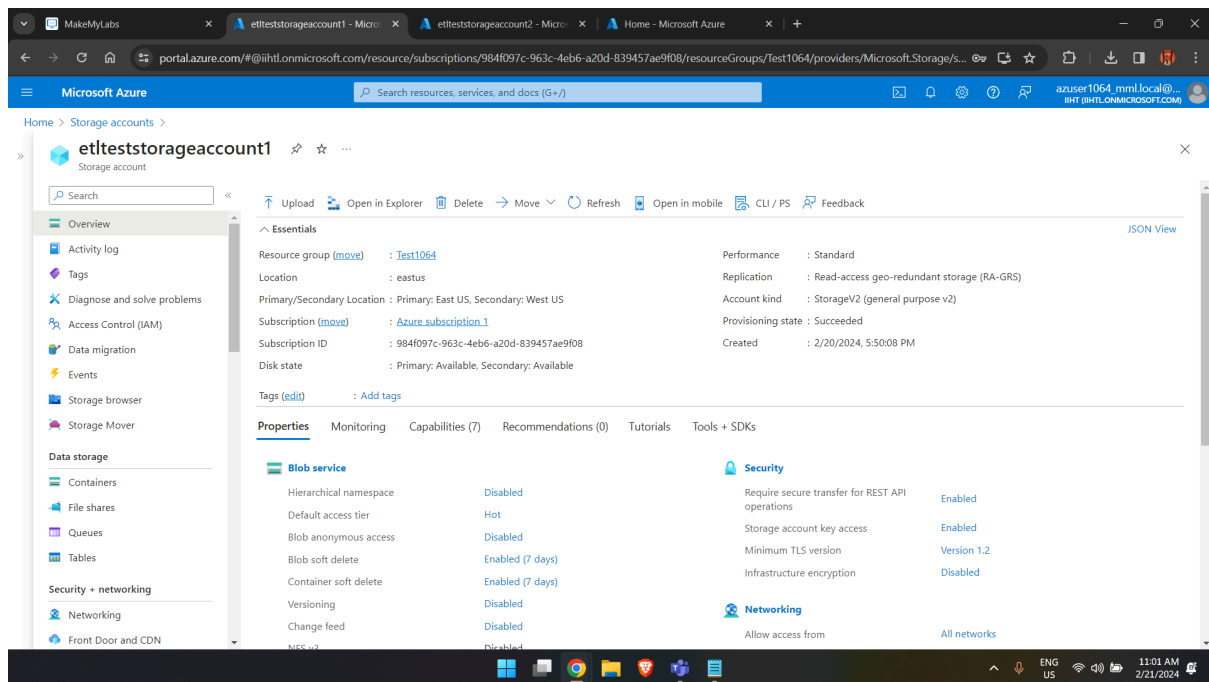
Azure Data Factory supports three types of integration runtimes: Azure, Self-hosted, and Azure-SSIS (SQL Server Integration Services).

Azure Data Factory simplifies the process of building, deploying, and managing data integration workflows in the cloud, enabling organizations to efficiently collect, process, and analyze data from diverse sources to gain valuable insights and drive business decisions.

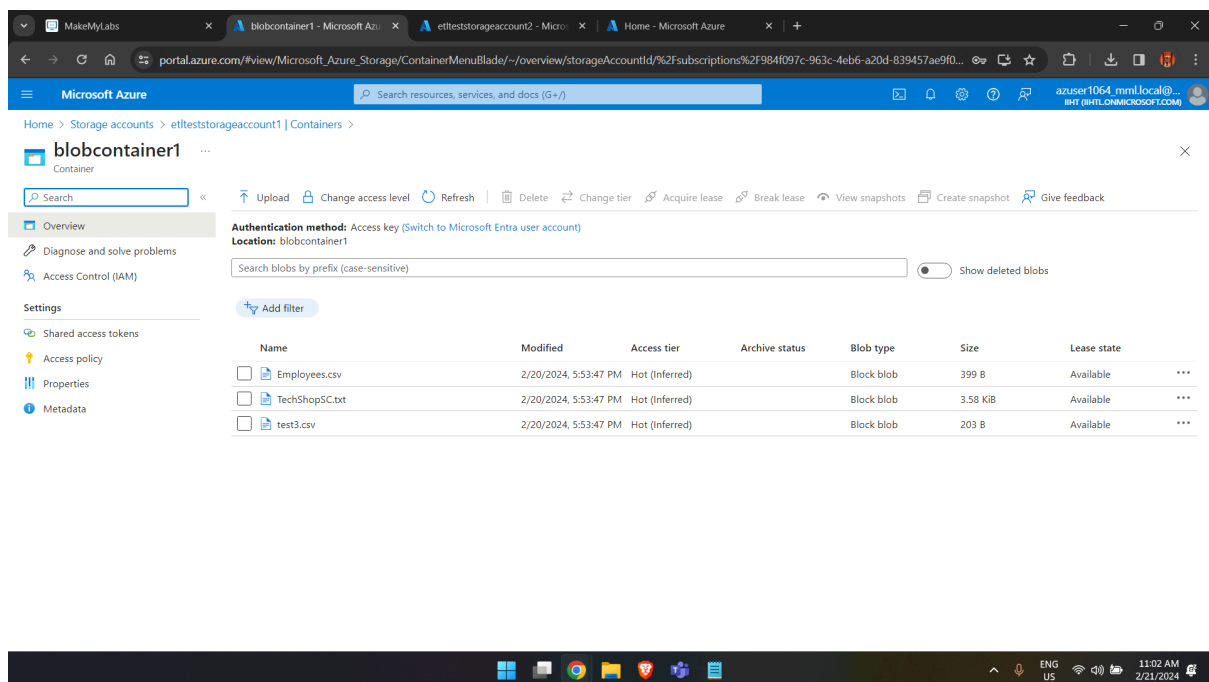
Executing Copy Activity :

Blob to Blob Copy :

Creating two blob storage accounts :

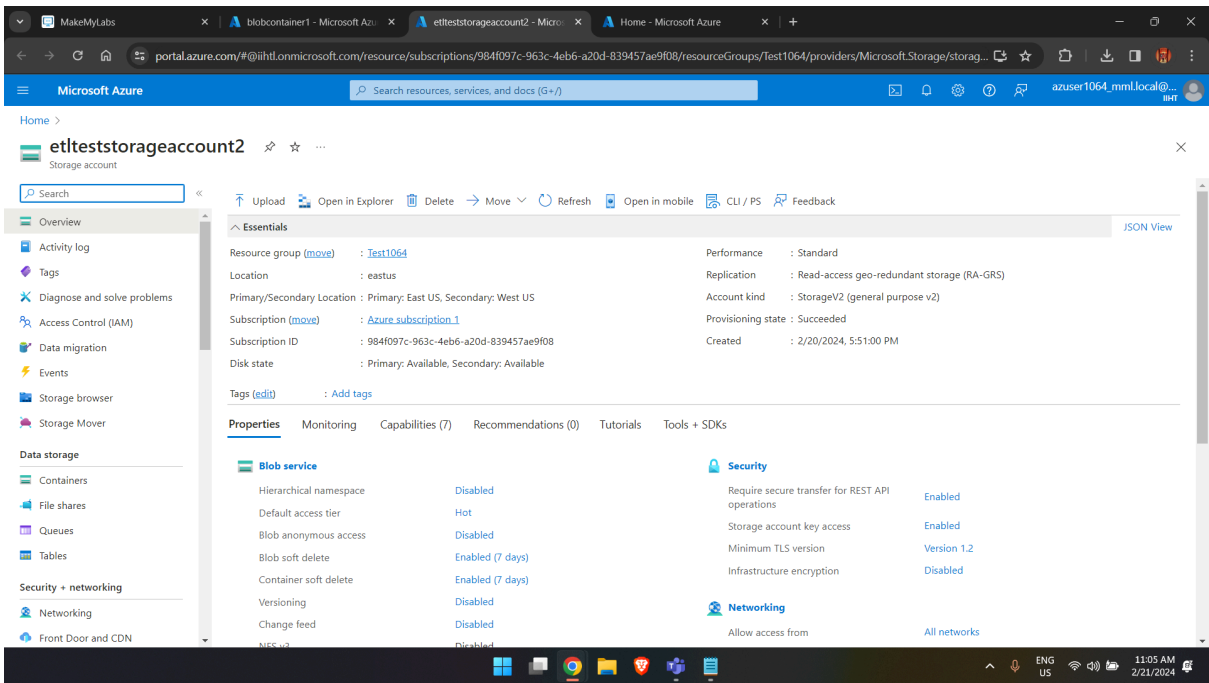


This account is for copying purposes so we will add sample data files in it.

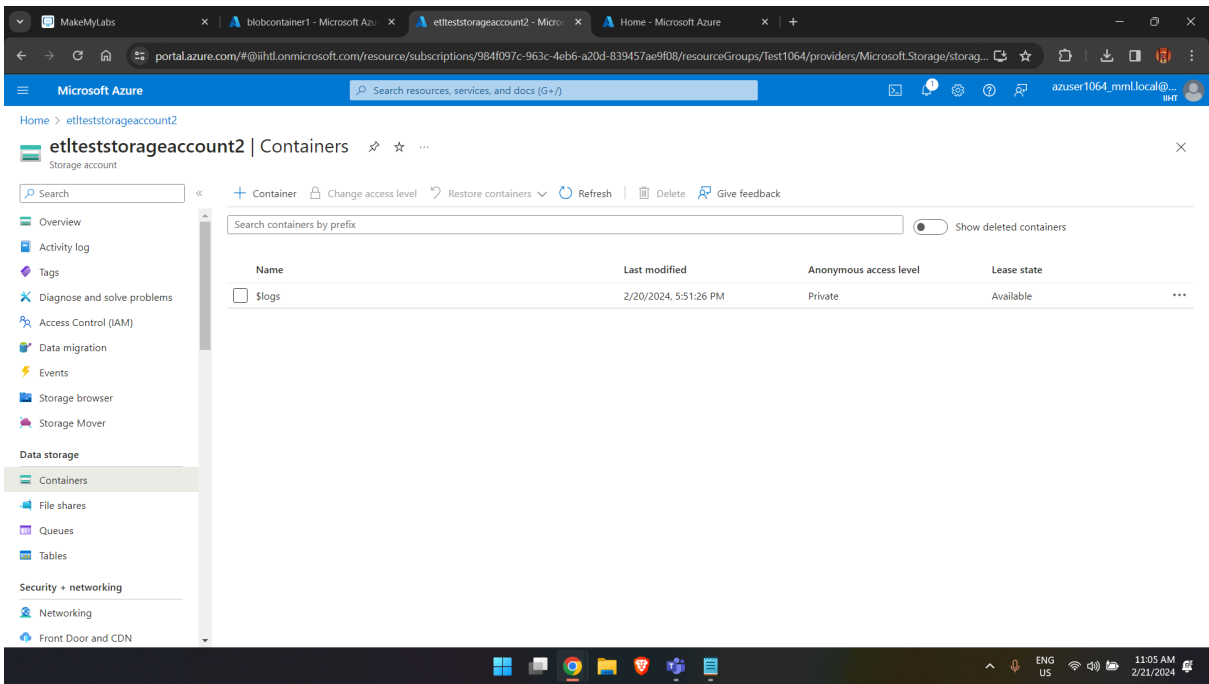


Added the sample data in blobcontainer1. Three files are added as we can see.

Creating a second blob storage account for copying data to it.



Storage account2 is created.



Here there is no container.

Creating new Data Factory :

Microsoft Azure

Home > Data factories >

Create Data Factory

Basics Git configuration Networking Advanced Tags Review + create

One-click to create data factory with sample pipeline and datasets. Try it

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 *

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

[Give feedback](#)

Created a datafactory.

Microsoft Azure

Home >

Microsoft.DataFactory-20240221110717 | Overview

Deployment

Search

Delete Cancel Redeploy Download Refresh

Overview

Your deployment is complete

Deployment name : Microsoft.DataFactory-20240221110717 Start time : 2/21/2024, 11:08:46 AM

Subscription : Azure subscription 1 Correlation ID : fd857d87-e240-41bd-9c16-cc22b28021b1

Resource group : Test1064

Deployment details

Next steps

[Go to resource](#)

Give feedback

[Tell us about your experience with deployment](#)

Cost management

Get notified to stay within your budget and prevent unexpected charges on your bill.

[Set up cost alerts >](#)

Microsoft Defender for Cloud

Secure your apps and infrastructure

[Go to Microsoft Defender for Cloud >](#)

Free Microsoft tutorials

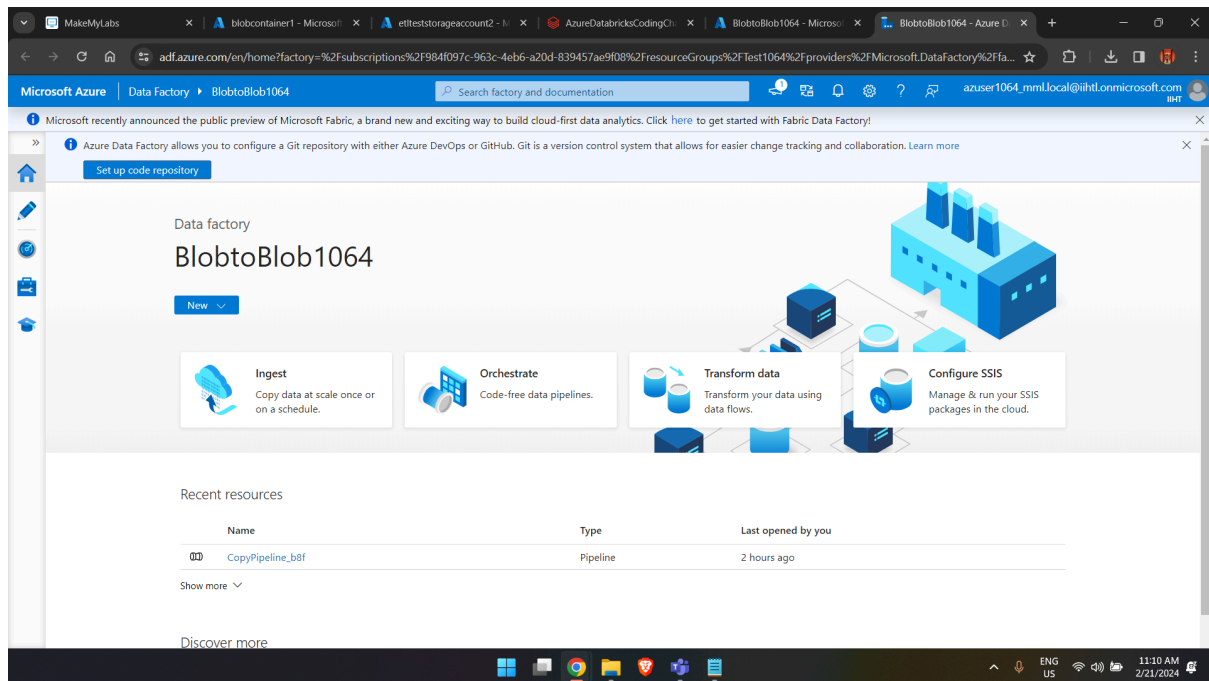
[Start learning today >](#)

Work with an expert

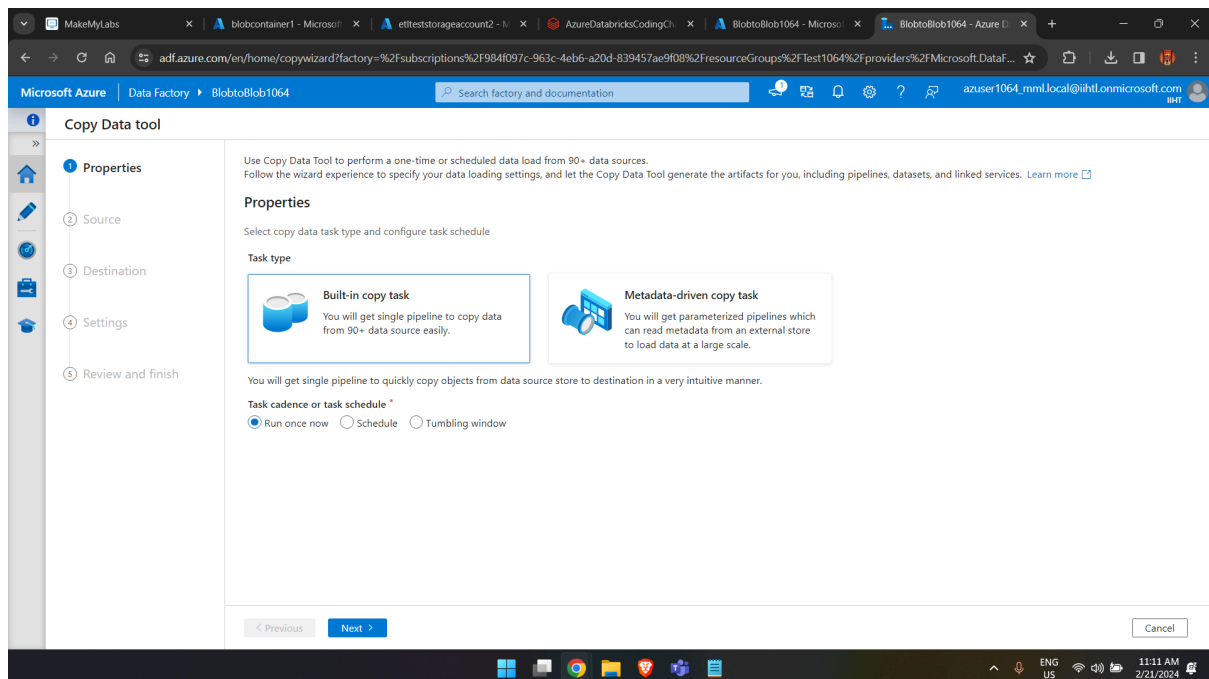
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.

[Find an Azure expert >](#)

After launching the studio. UI of studio will be generated .



Performing Copy Task:



Connecting the source file :

The screenshot shows the 'Copy Data tool' interface in the Microsoft Azure Data Factory portal. The 'Source data store' section is active, and the 'New connection' pane is open on the right. The 'Name' field is set to 'AzureBlobStorage1'. The 'Description' field is empty. The 'Connect via integration runtime' is set to 'AutoResolveIntegrationRuntime'. The 'Authentication type' is set to 'Account key'. The 'Account selection method' is set to 'From Azure subscription'. The 'Azure subscription' is set to 'Azure subscription 1 (984f097c-963c-4eb6-a20d-839457ae9f08)'. The 'Storage account name' is set to 'etiteststorageaccount1'. The 'Additional connection properties' section is empty. The 'Create' button is visible at the bottom of the 'New connection' pane.

Connecting Destination Storage Account :

The screenshot shows the 'Copy Data tool' interface in the Microsoft Azure Data Factory portal. The 'Destination data store' section is active, and the 'New connection' pane is open on the right. The 'Authentication type' is set to 'Account key'. The 'Account selection method' is set to 'From Azure subscription'. The 'Azure subscription' is set to 'Azure subscription 1 (984f097c-963c-4eb6-a20d-839457ae9f08)'. The 'Storage account name' is set to 'etiteststorageaccount2'. The 'Additional connection properties' section is empty. The 'Test connection' section is visible, with the 'To linked service' radio button selected. The 'Create' button is visible at the bottom of the 'New connection' pane.

Adding Destination Path :

Copy Data tool

Destination data store

Specify the destination data store for the copy task. You can use an existing data store connection or specify a new data store.

Destination type

Connection * [Edit](#) [+ New connection](#)

Folder path *
If the identity you use to access the data store only has permission to subdirectory instead of the entire account, specify the path to browse.
 [Browse](#)

File name

Copy behavior

Max concurrent connections

Block size (MB)

Metadata [+ New](#)


[< Previous](#) [Next >](#) [Cancel](#)

Reviewing the process :

Copy Data tool

Summary

You are running pipeline to copy data from Azure Blob Storage to Azure Blob Storage.



Properties [Edit](#)

Task name CopyPipeline_b5a

Task description

Source [Edit](#)

Connection name AzureBlobStorage1

Dataset name SourceDataset_b5a

Column delimiter ,

Escape character \

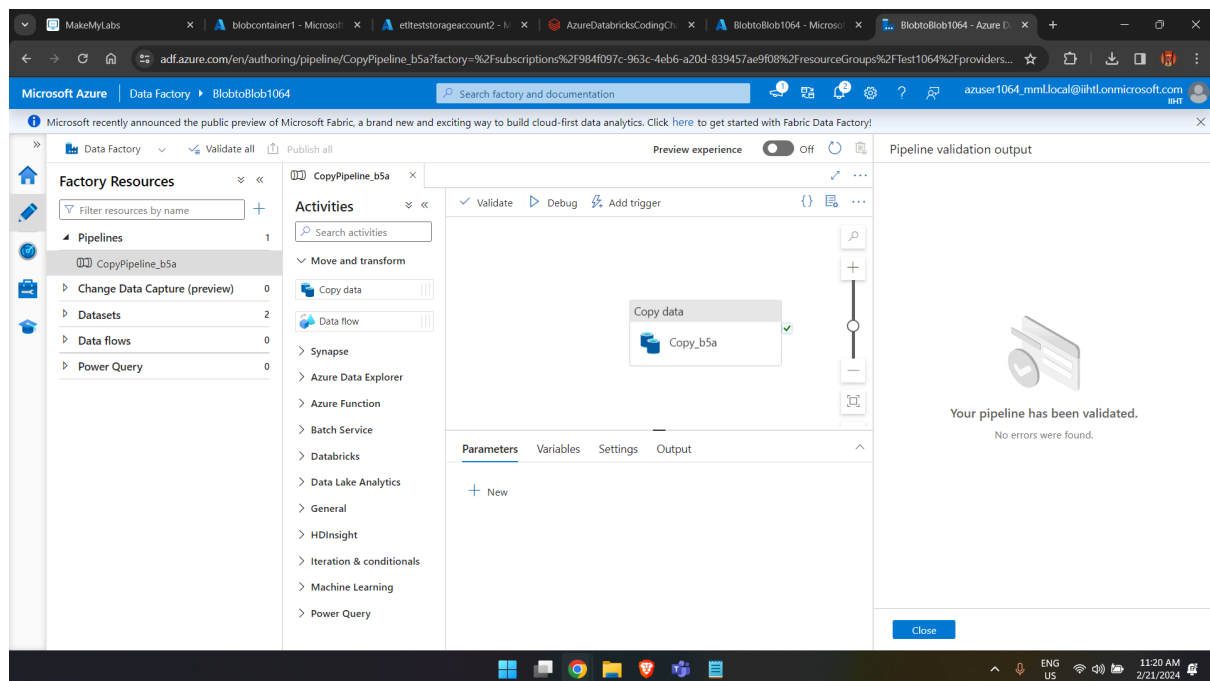
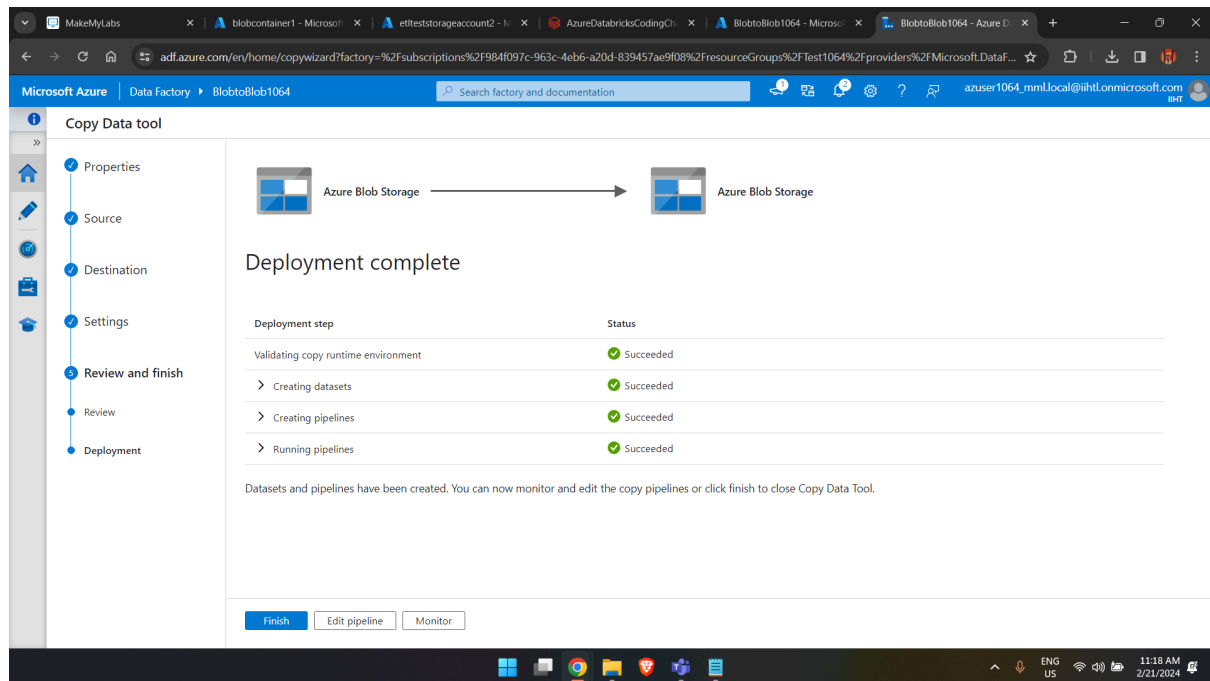
First row as header false

File name TechShopSC.txt

Container hlnhrrcontainer1

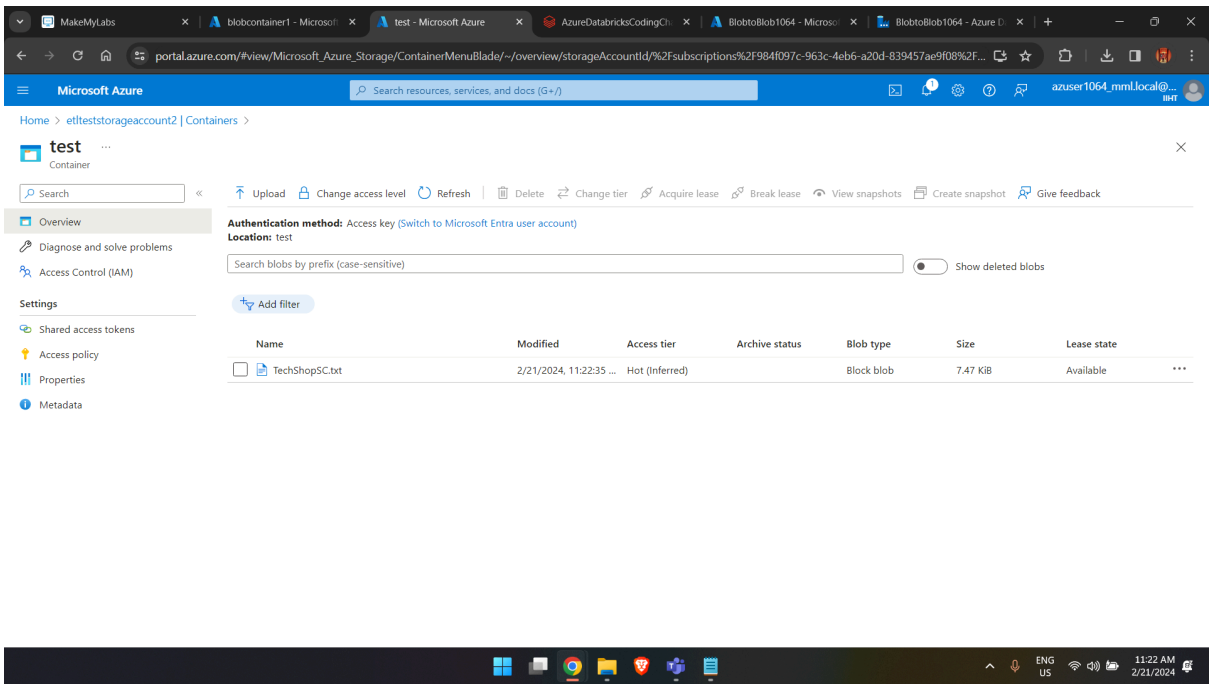
[< Previous](#) [Next >](#) [Cancel](#)

Creating pipeline :



Above Screenshot tells that the pipeline worked successfully.

Checking whether changes are reflected in storage account2 or not.



We can see the file is copied successfully from one blob storage account to another blob storage account.

From source to destination the data copied successfully by using azure data factory like that we can use this service for adls-adls, adls-blob, blob-adls.